CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Preference inference based on lexicographic and Pareto models
Authors	George, Anne-Marie
Publication date	2019
Original Citation	George, A-M. 2019. Preference inference based on lexicographic and Pareto models. PhD Thesis, University College Cork.
Type of publication	Doctoral thesis
Rights	© 2019, Anne-Marie George. - http://creativecommons.org/licenses/by-nc-nd/3.0/
Download date	2024-04-23 07:03:24
Item downloaded from	https://hdl.handle.net/10468/7781

UCC
University College Cork, Ireland
Coláiste na hOllscoile Corcaigh

Preference Inference Based on Lexicographic and Pareto Models

Anne-Marie George

MSc Mathematics

NATIONAL UNIVERSITY OF IRELAND, CORK

Faculty of Science

Department of Computer Science

Thesis submitted for the degree of Doctor of Philosophy

February, 2019

Head of Department: Prof. Cormac J. Sreenan

Supervisors: Dr. Nic Wilson
Prof. Barry O'Sullivan

Research supported by Insight Centre for Data Analytics and Science Foundation Ireland under Grant No. 12/RC/2289 which is co-funded under the European Regional Development Fund.

Contents

List of Figures iv
List of Tables v
List of Definitions vi
List of Theorems viii
List of Algorithms ix
Acknowledgements xii
Abstract xiii

1 Introduction **1**
1.1 Outline And Contributions 4
1.2 Publications 7

2 Related Work **9**
2.1 Preferences in Artificial Intelligence 9
2.1.1 Multi-Agent Decision Problems 10
2.1.2 Preferences for Single Agents 11
2.2 Binary Preference Relations 13
2.2.1 Relational vs. Cardinal Preference Representations . . . 14
2.3 Acquisition of Preference Information 16
2.3.1 Eliciting Preferences 17
2.3.2 Learning Preferences 20
2.3.3 Inferring Preferences 21
2.4 Preference Languages 22
2.4.1 Semantics 24
2.4.2 Types of Comparative Statements 27
2.5 Compact Representations of Preference Relations 30
2.5.1 Aggregation Functions 32
2.5.2 Qualitative Preference Models 37
2.6 Summary 43

3 Preliminaries **44**
3.1 Preference Inference 44
3.2 The Languages $\mathcal{L}^{\mathcal{A}}$, $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ 48
3.3 Pareto Models 51
3.4 Variable Hierarchies 56

4 Strong Compositionality **61**

4.1 Composition and Extension 62
4.2 Inference and Strong Compositionality 68
4.2.1 The Induced Relation $\models^*$ and Maximal Models 68
4.2.2 (Strongly) Compositional Preference Statements 71
4.2.3 Consistency of Strongly Compositional Preferences . . . 74
4.2.4 An Algorithmic Approach for the Consistency Problem with Strongly Compositional Statements 75
4.2.4.1 Minimum Model and Minimal Extensions . . . 76
4.2.4.2 The Algorithm 77
4.2.4.3 Minimal Model 78
4.2.5 Decreasing Preference Statements and Sets of Models . 79
4.3 Examples for Specific Model Types 82
4.3.1 Models and Composition Based on Unions of Variables . 82
4.3.2 CVO Lexicographic Models 86
4.3.2.1 Statements on Complete Variable Assignments 87
4.3.2.2 Statements $\varphi^{\mathcal{R}}$, Strict Versions and Negations . 90
4.3.2.3 Statements on Partial Variable Assignments . . 95
4.3.3 CVO Singleton Pareto Models 98
4.3.3.1 Statements on Complete Variable Assignments 98
4.3.3.2 Statements on Partial Variable Assignments . . 102
4.4 Discussion 105

5 Pareto Model **108**
5.1 Preliminaries 109
5.2 Properties and Solutions 112
5.2.1 Singleton Models 113
5.2.2 General Pareto Inference 116
5.3 Conclusion 121

6 FVO Lexicographic Model **122**
6.1 Preliminaries 123
6.2 Inconsistency Bases 124
6.3 Towards a Polynomial Algorithm 129
6.3.1 Γ-allowed sequences, i.e., models of $\Gamma^{(\geq)}$ 130
6.3.2 Maximal Γ-allowed sequences 134
6.3.3 The Algorithm 138
6.3.4 The case of inconsistent Γ 140
6.3.5 Orderings on variables 140

6.4 Strong Consistency and $\models^{max}$-Inference . . . 141
6.4.1 Max-model inference . . . 142
6.4.2 Properties of strong consistency and the associated inference . . . 145
6.5 Proof Theory for $\mathcal{H}(1)$-Inference . . . 147
6.5.1 Syntactic deduction $\vdash$ and soundness of inference rules . . . 148
6.5.2 Completeness of Proof Theory . . . 152
6.6 Discussion . . . 154

7 Hierarchical Model **157**
7.1 Preliminaries . . . 158
7.2 coNP-completeness of $\mathcal{H}(t)$-Deduction for $t > 1$ . . . 164
7.3 Solving $\mathcal{H}(t)$-Consistency with $t > 1$ . . . 171
7.3.1 MILP Formulation . . . 172
7.3.2 Recursive Algorithms . . . 175
7.4 Experimental Runtime Comparisons . . . 183
7.5 Discussion . . . 187

8 CVO Lexicographic Model **189**
8.1 $\mathscr{L}$-Consistency for $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ . . . 190
8.1.1 Projections to Y . . . 191
8.1.2 Checking Γ-Satisfaction . . . 193
8.1.3 $\models^*$-Models for Subsets of $\mathcal{L}'_{pqT}$ . . . 197
8.1.4 The Algorithm . . . 201
8.2 Optimal Alternatives . . . 203
8.2.1 Notions of Optimality . . . 203
8.2.2 Optimality for $\mathscr{L}$ and Compositional Statements . . . 205
8.2.3 Computing Optimal Solutions for $\mathscr{L}$ and $\mathcal{L}'_{pqT}$ . . . 209
8.2.3.1 Membership Tests . . . 210
8.2.3.2 Incremental Approaches . . . 210
8.3 Discussion . . . 216

9 Conclusion **217**
9.1 Summary . . . 217
9.2 Possible Future Work . . . 219

List of Figures

7.1 Mean running times in seconds of PC-check($\mathcal{V}, \Gamma, \oplus, t$) (left) and PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) (right). 185

List of Tables

7.1 Mean running times in seconds to solve PCP with the MILP formulation over 25 instances and with PC-check over 50 instances. 185

7.2 Mean times of PC-check in seconds fixing $m = 25$ and ratios of the mean times between PC-check($\mathcal{V}, \Gamma, \oplus, t$) and PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) rounded to the nearest hundredth. *Mean time over 5 instances only. (All remaining are mean times over 50 instances.) 186

List of Definitions

2.1 Preference Relation 13
2.2 Order Relations 13
2.3 Equivalence, Strict and Incomparability Relations 14
2.4 Lexicographic Order 15
2.5 Pareto Order 15
3.1 Preference Deduction Problem 46
3.2 Preference Consistency Problem 47
3.3 The Language $\mathcal{L}^{\mathcal{A}}$ 48
3.4 Non-Strict Versions of Statements $\mathcal{L}^{\mathcal{A}}$ 48
3.5 The Language $\mathcal{L}_{pqT}$ 50
3.6 Tuples of Extending Alternatives for Statements in $\mathcal{L}_{pqT}$ 50
3.7 The Language $\mathcal{L}'_{pqT}$ 51
3.8 Fixed-Value-Order (FVO) Singleton Pareto Models 52
3.9 FVO Singleton Pareto Order 52
3.10 k-bound Pareto Models 53
3.11 k-bound Pareto Order 53
3.12 Changeable-Value-Order (CVO) Singleton Pareto Models 56
3.13 CVO Singleton Pareto Order 56
3.14 Fixed-Value-Order (FVO) Lexicographic Model 57
3.15 FVO Lexicographic Order 57
3.16 Hierarchical Models 57
3.17 Hierarchical Order 58
3.18 k-bound Hierarchical Models 58
3.19 Changeable-Value-Order (CVO) Lexicographic Models 60
3.20 CVO Lexicographic Order 60
4.1 Composition Operator 62
4.2 Extension Relation 63
4.3 $\models^*$-Satisfaction Relation 68
4.4 Maximal Models 69
4.5 Maximal $\models^*$-Models 70
4.6 (Strong) Compositionality 71
4.7 Minimum Models 76
4.8 Minimal Extensions 77
4.9 Minimal Models 78
4.10 Decreasing Statements 80
4.11 Decreasing Sets 81

4.12 Containing Models of Statements 81
4.13 Relaxations of Statements 82
4.14 Variable Mappings 83
4.15 Maximal Models Variable Set V^{Γ} 85
4.16 Maximal Model Inference 85
4.17 Strict Versions of Statements 91
4.18 Variable Projections 92
4.19 Simultaneous Decisiveness 95
5.1 Reversed Statements 111
6.1 Inconsistency Base 125
6.2 Γ-Allowed Sequences 130
6.3 Maximal Γ-Allowed Sequences 134
6.4 Strong $\mathcal{H}(1)$-Consistency 142
6.5 Pareto Difference Relation 147
7.1 Hierarchical Structures 158
8.1 $\text{Best}^{\pi}_{\Gamma}(X)$, $\text{Worst}^{\pi}_{\Gamma}(X)$ and $\text{Pairs}^{\pi}_{\Gamma}(X)$ 197
8.2 Variables That Can Be Chosen Next 198
8.3 Valid Extensions 199
8.4 Optimality Operators 211

List of Theorems

4.1 Maximal Model Satisfaction of Preferences 74
4.2 (Strong) Compositionality of $\mathcal{L}_{pqT}$ and Negations 97
5.1 $\mathcal{P}(1)$-Consistency 113
5.2 $\mathcal{P}_{\mathcal{V}}(1)$-Deduction 115
5.3 NP-completeness of PCP for Pareto Models 118
5.4 coNP-completeness of PDP for Pareto Models 120
6.1 Correctness of the Algorithm 139
6.2 Completeness of Proof Theory (1) 153
6.3 Completeness of Proof Theory (2) 154
7.1 Mutual Expressiveness of Consistency and Deduction 162
7.2 coNP-completeness of PDP for Hierarchical Models 164
7.3 NP-completeness of PCP for Hierarchical Models 170
7.4 Singleton Set Sequences in Γ-Satisfying Models 179
7.5 Conflicting Sets 181
8.1 Γ-Satisfaction of $\models^*$-Model 196
8.2 Set Relations of Different Optimality Classes 209

List of Algorithms

4.1 Consistency for Strongly Compositional Statements Γ 77

5.1 Singleton-Pareto-Consistency(Γ,$\mathcal{V}$) 114
5.2 Singleton-Pareto-Deduction(Γ,$\mathcal{V}$,φ) 115

6.1 $\mathcal{H}(1)$-Consistency for Statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ 138

7.1 PC-check($\mathcal{V}, \Gamma, \oplus, t$,) 180
7.2 PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S} = \emptyset, s$) 182

8.1 $\mathscr{L}$-Consistency for Statements $\Gamma \subseteq \mathcal{L}'_{pqT}$ 201
8.2 Computing $\Omega(A)$ for Optimality Operator Ω 212
8.3 Computing $N(A)$ when (I′)-(V′) holds 213

I, Anne-Marie George, certify that this thesis is my own work and I have not obtained a degree in this university or elsewhere on the basis of the work submitted in this thesis.

Anne-Marie George

To my parents.

Acknowledgements

First and foremost, I would like to thank Nic Wilson, my PhD supervisor, for his constant and valuable support, patience and encouragement. I am very grateful for his excellent supervision that made the process of getting a PhD enjoyable and insightful. I am also thankful for the support of my second supervisor Barry O'Sullivan and for our fruitful discussions. Furthermore, I would like to thank the examiners of this thesis for their comprehensive evaluations as well as other reviewers of the here presented publications. I very much appreciated the technical and administrative support of the Insight UCC staff, as well as opportunities for outreach and teaching activities. I am grateful for the financial support of the Insight Centre for Data Analytics and Science Foundation Ireland by Grant No. 12/RC/2289 which is co-funded under the European Regional Development Fund.

Finally, I would like to thank my family, and in particular my parents, who always gave me support, advice and strength. I am grateful to all loved ones, friends and colleagues that accompanied me with understanding and cheerfulness through my PhD and other adventures.

Abstract

Preferences play a crucial part in decision making. When supporting a user in making a decision, it is important to analyse the user's preference information to compute good recommendations or solutions. However, often it is impractical or impossible to obtain complete knowledge on preferences. Preference inference aims to exploit given preference information and deduce more preferences. More specifically, the Deduction Problem asks whether another preference statement can be deduced from a given set of preference statements. The closely related Consistency Problem asks whether a given set of user preferences is consistent, i.e., the statements are not contradicting each other.

We present approaches for preference inference based on qualitative preference models that are based on lexicographic and Pareto orders. We consider user preference statements that are given in the form of comparisons of alternatives or alternative sets. For some model types and preference statements we formulate efficient algorithms; for others we show NP-completeness and coNP-completeness results. In particular, we find that the Deduction and Consistency problem are polynomial time solvable for comparative preference statements for lexicographic and simple Pareto preference models by a detailed analysis of the problem structures. The computational efficiency for these models makes them particularly appealing for practical uses. The Deduction and Consistency Problem are coNP-complete and NP-complete, respectively, for hierarchical and generalised Pareto models, which make these models less practical even for simple preference languages. However, we still formulate a quite efficient algorithmic approach to solve the Consistency Problem (and implicitly the Deduction) for hierarchical models.

We also analyse deduction and consistency for preference statements that are (strongly) compositional under some set of preference models. (Strong) compositionality is a property of preference statements in connection with a set of preference models. It demands inference of preference statements for certain combinations of preference models. We find many interesting results for this case, which ultimately leads to a general greedy algorithm to solve the Consistency Problem for strongly compositional preference statements. This indicates that strong compositionality is an important property that can deliver immediate algorithmic approaches when present. We find many types of preference statements, e.g., conjunctions of strongly compositional statements, are

strongly compositional. The considered comparative preferences statements are also strongly compositional for many of the discussed preference models - different lexicographic and hierarchical models.

We can make use of the Deduction Problem to find a set of optimal alternatives, e.g., to be recommended to a user that are undominated with respect to different notions of optimality. We analyse this connection for general lexicographic models and find computationally efficient solutions.

Chapter 1

Introduction

The Cambridge English Dictionary's first definition of the word preference is: "the fact that you like something or someone more than another thing or person" [Dic18]. Following this definition, preferences play a crucial role in any decision making process for a user when presented with a choice of items, candidates or alternatives. As such, they are used in various fields in computer science, philosophy, economics and operations research. In particular, applications in artificial intelligence can be found: in voting theory for choosing a winner, in matchings to find stable solutions, in data bases to refine search queries, in recommender systems to present a user with the most appreciated alternatives, in human-computer interaction to optimise the use of computer interfaces, in multi-objective (constraint) decision making to find an optimal solution, and others [BD09, DHKP11]. Since preferences vary depending on the user's origin, religion, taste, social environment, requirements, etc., they are personal and must be elicited from the user directly when used for supporting a user in a decision, or learned from previous decisions.

The utilisation of preferences is accompanied by various issues. This PhD work takes on the task of analysing the structure, determining the complexity and designing efficient algorithms (where possible) for the following interconnected problems:

- Given a set of user preference statements Γ, are these statements consistent, i.e., do not contradict each other? (Consistency Problem)
- Given a set of user preference statements Γ and a query preference statement φ, can we deduce that φ holds true for the user as well? (Deduction Problem)

- Given a set of user preference statements Γ, what are optimal alternatives to present to the user? (Optimal Alternatives Problem)

We explore these problems for known and newly designed qualitative preference models that are based on lexicographic and Pareto orders, and comparative preference statements. While we find that all three problems are coNP-complete and NP-complete, respectively, for hierarchical and generalised Pareto models, Deduction and Consistency are solvable in fast polynomial times for types of lexicographic models and simple Pareto models. Many of our algorithms for the polynomial time cases are greedy approaches and rely on the underlying structure of the problems and model types. One particularly interesting property that we find is (strong) compositionality. We show that the Consistency Problem can be solved by a greedy approach for any strongly compositional input statements. We consider the Optimal Alternatives Problem for different notions of optimality for general lexicographic models and find computationally efficient solutions.

An elicitation of the exact preferences of a user is often unrealistic due to the immense effort required for the user to rank a large amount of alternatives. Furthermore, a user might not be concious about their exact preferences, e.g., due to the lack of information on alternatives or the difficulty of expressing the importance of features in numerals. It is thus often preferred to ask the user only few and relatively simple questions about her or his preferences. A fitting preference language has to be chosen accordingly. Often, a tradeoff between the expressiveness of the preference language and the convenience for the user has to be found. It is, for example, much easier for a user to compare "I prefer hotel A to hotel B" rather than to state "I prefer any hotel close to a beach 3.5 times more to any 5 star hotel that has a pool and is far from the beach". However, while the first statement only specifies the relation of two alternatives, the latter is much more expressive since it orders many pairs of alternatives and gives a measure of "how much" alternatives are preferred over others. Apart from choosing an appropriate preference language, one can also consider which questions to ask to the user in order to obtain maximal information. In this dissertation, we mainly focus on problems of preference handling under different types of comparative preference languages. That is, the user gives quantitative comparisons of alternatives or alternative sets.

Another problem that needs attention when handling preferences is that elicited preferences might not be consistent, meaning they contradict each other. If

preferences are stated over a longer time period, one approach to overcome inconsistency can be to consider only the latest consistent set of preference statements. In other applications, it is reasonable to handle inconsistency by finding an approximation that fits the true preference model for a user best. This learning of preference models is closely related to preference elicitation.

Under many assumptions on the user's preference model, deciding consistency is related to inference of new preferences. Inference of preferences is used to deduce more preference information from a given set of preferences in order to overcome a possible gap of knowledge which could, for example, originate from eliciting only few preferences from the user for her or his convenience.

In order to simplify and strengthen the analysis of consistency and inference for preferences, assumptions can be made on the user's preference model, i.e., assumptions are made on the way the user expresses her or his preferences. Preference learning techniques focus on finding one preference model of the assumed type that agrees best with the given preferences [FH10a]. This overcomes the inconsistency problem, but only gives an approximation of the real user preference model which can lead to wrong deductions. Other work concentrates on reasoning with the set of all preference model of the assumed type that satisfy the given preferences, see [Wil14] and our work [GW16, WGO15, WG17]. This set clearly must include the true user preference model. Thus, no false deductions can be made (assuming that the true preference model is of the assumed type). However, in the case of inconsistent preference statements, the set of all preference models of the assumed type that satisfy the given preferences is empty, and no meaningful deductions of other preferences can be made.

In this dissertation, we will concentrate on the latter approach of reasoning with the set of all preference model of the assumed type that satisfy the given preferences. We develop characterisations of consistency and deduction for different assumptions on the type of preference models and for different preference languages. Furthermore, we develop algorithms and complexity results to solve these problems and compare some of their running times.

To present users with optimal choices or solutions in a multi-objective framework based on their preferences, one needs to find an appropriate measure of optimality. One of the simplest methods is to assign an importance parameter to every objective and use a weighted sum of objectives to identify optimal solutions [FGE05]. However, if preference statements on solutions are given, one

can reason over a set of preference models that satisfy the given preferences. Every such preference model induces an order relation on the set of solutions. We can define an optimality operator (an operator on alternative sets that returns a set of "optimal" alternatives) based on a set of order relations. Similarly as in a voting scenario one needs to decide on a rule by which a winner/set of winners is selected. In this dissertation, we consider many known notions of optimality [WO11] for one specific type of preference model and compare the relations of these operators towards each other. Furthermore, we develop algorithms to compute these operators and consider their complexity.

1.1 Outline And Contributions

The contribution of this work is a detailed analysis of preference deduction and preference consistency for some qualitative preference models that are based on the structures of well-known order relations: Lexicographic and Pareto orders. The aim is to investigate necessary and sufficient conditions for deduction and consistency. In the course of this analysis, we develop simple algorithms to solve consistency and deduction for some cases, and prove NP-completeness and coNP-completeness results for other cases.

This dissertation is organised in the following way.

Chapter 2: Related Work This chapter gives a brief overview of related work for preferences in areas of artificial intelligence. We present the general concept of preference relations. Furthermore, an overview of some preference elicitation and learning techniques is given together with issues that arise in these approaches. Afterwards, we discuss different representations of preferences. We present commonly used preference languages and compact representations (preference models).

Chapter 3: Preliminaries In this chapter, we outline all important preliminaries for the framework used in the remainder of the dissertation. Here, we formalise the Preference Consistency and Preference Deduction Problem. We then introduce all specific preference languages and associated notations that this dissertation is focusing on. The novel ideas of Pareto models and Hierarchical models are formally defined together with their induced order relations

on alternatives and other important notions.

Chapter 4: Strong Compositionality This chapter analyses consistency and deduction of preferences in very general conditions. No specific model type or preference language is assumed here (although examples are extensively discussed). Instead we develop characterisations and conditions for deduction and consistency based on (strongly) compositional preference statements. Here, (strong) compositionality is a property of preference statements in connection with preference models that is based on a composition operator by which preference models can be combined. We show that a greedy approach can be applied to solve the Consistency Problem for strongly compositional statements and can be efficiently implemented depending on the type of preference model. This greedy approach and its complexity is described in more detail in many other parts of this dissertation for different preference models. Furthermore, we discuss some interesting examples of strongly compositional preference statements for some types of lexicographic and Pareto models.

Chapter 5: Pareto Model Here, we focus our analysis of deduction and consistency on Pareto models for strict and non-strict comparative statements on alternatives. In contrast to the hierarchical/lexicographical models considered in this thesis, deduction and consistency are not mutually expressive for Pareto models. This means that one problem cannot be solved by solving the other, and thus, consistency and deduction are considered separately. We first consider the case where Pareto models include only singleton variable sets, i.e., alternatives are compared in a Pareto manner based on the variables included in the model. Here, we design polynomial algorithms based on simple set relations for both the Deduction and the Consistency Problem. For Pareto models that can include non-singleton sets of variables, the values of variables within one set are first aggregated by an operator, and alternatives are then compared in a Pareto manner based on these aggregated values. For this type of models, the Deduction and Consistency Problem are proven to be coNP-complete and NP-complete, respectively.

Chapter 6: FVO Lexicographic Model Fvo lexicographic models are preference model in which variables have a strict importance order. Alternatives are compared lexicographically, i.e., on their values of the most important variable,

and only if these values are equal are they compared on the second most important variable, and so on. The Deduction and Consistency Problems, which are mutually expressive for the simple form of fvo lexicographic models considered in this chapter, can be solved in low-order polynomial time. Apart from presenting an algorithm to solve these problems for strict and non-strict comparative preference statements, we also analyse their structure. We describe an interesting concept, inconsistency bases, which lead us to many helpful properties of models that satisfy preference statements. We can state that the algorithm, even in the case of inconsistent preference statements, finds the "most satisfying" preference model and also identifies a maximal set of satisfiable preference statements. Furthermore, we are able to use the same algorithm to deal with comparative preference statements on the importance of variables. We also consider strong consistency, which asks if there exists a model that satisfies the given user preferences and includes all variables. Finally, we develop a proof theory and show its completeness. This shows that we are able to consider this form of preference inference not only from a semantic definition but also from a logical perspective.

Chapter 7: Hierarchical Model In this chapter, we consider hierarchical preference models, in which variables are ordered by importance, but can be equally important as well. The values of variables within one set of equally important variables are first aggregated by an operator, and alternatives are then compared in a lexicographic manner based on these aggregated values. We show for strict and non-strict comparative preference statements that the Deduction and Consistency Problem, while being mutually expressive, are coNP-complete and NP-complete, respectively. We then concentrate on finding efficient approaches to solve the Preference Consistency Problem (and thus the Preference Deduction Problem). We give a Mixed Integer Linear Programming formulation, and then focus on recursive search algorithms that explicitly exploit some properties of the problem that have been developed. Afterwards, we give a description of runtime experiments to compare the efficiency of the described approaches.

Chapter 8: CVO Lexicographic Model Cvo lexicographic models like their simpler version of fvo lexicographic models, compare alternatives lexicographically based on a strict importance order of variables. However, here we assume that the order on the values of variables is not fixed but specified within the

model. We consider the Deduction and Consistency Problem, which again are mutually expressive, for these models in connection with preference statements that are certain comparisons over sets of alternatives. Even in this more general case (compared to our considerations on simple lexicographic models), we can formulate a polynomial time algorithm. We then consider different notions of optimality for a set of alternatives. Based on the developed algorithm for consistency, we show how (and how efficient) sets of optimal solutions can be computed for the different notions of "optimal".

Chapter 9: Conclusion In the conclusion, we summarise the work presented in this dissertation and point out important results. Furthermore, we outline possible future work.

1.2 Publications

Parts of Chapters 3- 8 are based on the following published papers, which have been subject to peer review:

- Nic Wilson, Anne-Marie George, and Barry O'Sullivan. Computation and complexity of preference inference based on hierarchical models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) 2015*, pages 3271–3277, 2015.
- Anne-Marie George, Abdul Razak, and Nic Wilson. The comparison of multi-objective preference inference based on lexicographic and weighted average models. In *Proc. International Conference on Tools with Artificial Intelligence (ICTAI) 2015*, pages 88–95, 2015.
- Anne-Marie George and Nic Wilson. Preference inference based on pareto models. In *Proc. International Conference on Scalable Uncertainty Management (SUM) 2016*, pages 170–183, 2016.
- Anne-Marie George, Nic Wilson, and Barry O'Sullivan. Towards fast algorithms for the preference consistency problem based on hierarchical models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) 2016*, pages 1081–1087, 2016.
- Nic Wilson, Anne-Marie George, and Barry O'Sullivan. Preference inference based on hierarchical and simple lexicographic models. *IfCoLog Jour-*

nal of Logics and their Applications, 4:1997–2038, 2017.

- Nic Wilson and Anne-Marie George. Efficient inference and computation of optimal alternatives for preference languages based on lexicographic models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI) 2017*, pages 1311–1317, 2017.

Chapter 2

Related Work

In this chapter, we give a survey on related work for the problems of preference consistency and deduction discussed in this dissertation. This includes the use of preferences in areas of artificial intelligence in general, and single-agent decision problems in particular. Afterwards we give a formalisation of preference relations. We then discuss related work in preference elicitation and learning. Several representations of preferences are described by giving details on commonly used preference languages and compact representations of preference models.

2.1 Preferences in Artificial Intelligence

Preferences are considered in different contexts in artificial intelligence. They usually involve either multiple agents (representing real users/persons), or multiple criteria over which preferences can be expressed. Preferences of multiple agents can be conflicting. Similarly, preferences over multiple criteria can be opposing. It is thus not obvious how to make decisions or find "optimal" solutions in such scenarios. This dissertation focuses (especially in this chapter) on single-agent decision making. However, in some of the presented work, the different criteria can be viewed as preferences of multiple agents towards the alternatives and are treated similarly as in voting scenarios.

Uncertainty in handling preferences can arise from a lack of information on user preferences, but also from uncertainty in the alternative's features. For example, the actual flight time of different flight options can vary due to delays.

Alternatives can then be compared on their expected values due to probability distributions [Fis70]. We concentrate our analysis on the former form of uncertainty, where insufficient preference information is given.

For the assumptions made on preferences in this dissertation, preference systems are monotonic. This means that a superset of preferences $\Gamma \cup \Phi$ yields a more constrained system, and the set of preferred alternatives A' will be a subset $A' \subseteq A$ of the set of preferred alternatives A derived from Γ. In contrast, preference handling approaches for non-monotonic systems assume that some preference information Γ may lead to some set of preferred alternatives A, while a superset of preferences $\Gamma \cup \Phi$ may lead to a different set of preferred alternatives [DSTW04].

2.1.1 Multi-Agent Decision Problems

Classical examples of multi-agent decision problems can be found in social choice theory for voting, matching and fair allocation problems [BCE+16].

A voting problem involves several voters that express preferences over several candidates. Based on some voting rule, a winner is selected among the candidates. Many voting rules have been considered in the computational social choice literature, mostly under the analysis of computational complexity and desirable properties, such as non-dictatorship, neutrality, unanimity, independence, etc. Some impossibility results have been shown for the construction of voting schemes that satisfy several properties simultaneously. More details on voting schemes are given later on (Subsection 2.5.1). An extensive introduction to voting theory can be found in [BCE+16].

Matching problems in general try to find pairings of objects. A bipartite matching problem, considers matching objects of one class to objects of another class. When considering users that are to be matched to other objects or users, it is only natural that preferences have to be considered. Considerations of "stability" often accompany matchings problems including preferences. Stability means that for a matching no feasible couple can be found which prefers to be matched to each other rather than with the assignment given by the matching. The Stable Marriage, Stable Rommate and Hospital Residence Problem and several variants, for example, apply this concept of stability to various application problems such as kidney donation schemes [Man13], centrally coordinated schemes for college admissions, school choice programmes, and allocation of

medical residents to hospitals [Bir17]. Here, graph theoretical properties can be exploited to solve these problems.

Fair allocation problems deal with problems of allocating or dividing divisible [BCM16] or indivisible [Pro16] goods between users. Indivisible goods are objects, e.g., houses, cars, paintings, etc. which cannot be divided. Fair allocation problems with divisible goods are often explained with the cake cutting problem. A cake as to be split among cake eaters, however, a cut can be made at any position. In this sense, matching problems form a type of allocation problems over indivisible goods. In general, different users might have strong preferences for different or the same goods. Here, the question of "fairness" arises. How can we divide goods in a fair way? Notions of fairness include Pareto optimality (efficient allocations) and envy-freeness [BCE+16]. Efficient allocations allocate or divide goods in a way that there exists no other allocation which is equally or more preferred by all agents, and strictly preferred by at least one agent. Envy-freeness expresses that no agent prefers another agent's share. Often fair allocation problems are analysed under the existence and properties of fair solutions as well as complexity results [Tho16].

Examples of multi-agent decision problems, where agents individual decisions influence the decisions of other agents, can be found in game theory [Osb04], planning and scheduling (e.g., in Nurse Roostering [BDCBVL04] and Personell Scheduling [VdBBDB+13]), argumentation theory [KVDT08, MP13], combinatorial auctions [DVV03], and others. Here, agents are usually autonomous, goal-oriented, and pro-active [Bul14], so that their preference usually is to maximise/minimise some objective.

2.1.2 Preferences for Single Agents

Many studies, like [BDSS15, HWB+11, IJH04, SS11], report that, if many options are available, a user's choice is likely to be poor (in comparison with objective optimal options). It is thus valuable to support users in their decision making when faced with a large set of options. This might be because of the high number of alternatives, but also because of the existence of several relevant features of the offered alternatives. Single-Agent Decision Problems usually involve multiple preference criteria or objectives.

Recommender systems are concerned with presenting a user with desirable available options. Major efforts have been made to understand user preferences

from a more psychological perspective. Users can be compared based on similar choices or behaviour [SK09], and a user profile can be constructed [LDGS11]. It is analysed what kind of recommendations a user values or expects. For example, some users might value serendipity, novelty, diversity and other properties in recommendation sets [KB17]. Furthermore, one can assume that preferences are not necessarily static, but can change over time [BA09], which raises the question of which time frame to consider. Furthermore, the same user can have different preferences over the same alternatives, depending on the situation the user is in [BA09]. This imposes some conditionality on preferences. Note that the preference relation on the criteria under which alternatives are considered can also depend on the user. While users usually prefer low prices to high prices, it might not be clear whether a user prefers a black phone to a silver one.

In multi-objective optimisation problems, variables represent the criteria of alternatives and have numerical domains, so that the order relations on the domains are given by natural orders. The set of available/feasible alternatives can sometimes be specified by a set of constraints on the variable domains. Multiple objectives/utilities express the overall goals of a user by which alternatives can be compared. The objective values or variable values for feasible alternatives can be aggregated into numerical values [FGE05], e.g., by (utility) functions, or compared by relational orders [FHWW10, Jun04, MRW13], e.g., Pareto frontiers or lexicographic orders. We discuss applicable order relations, by which a set of "optimal" solutions can be found, in more detail in Section 2.5.

User search queries in databases can be personalised by incorporating preferences. Research in this area is concerned with the representation of preferences in a database framework, and the complexity of finding answers to the search queries based on these preferences. Here, preferences are viewed as soft constraints, i.e., do not represent the primary orientation of the search. As such they help to avoid empty answer sets, and enable a ranking of answers, so that the user is not overwhelmed with two many answers [LL87]. Within the field of database systems, various preference representations have been analysed and implemented. A wide overview can be found in [Kie02] and [SKP11].

2.2 Binary Preference Relations

To understand what preferences are, we consider the definition of a binary relation $\trianglerighteq$ and important associated properties. A preference relation orders alternatives, i.e., sets alternatives into relation with each other, and can thus be formally defined as a binary relation on a set of alternatives $\mathcal{A}$.

Definition 2.1: Preference Relation

A *preference relation* $\trianglerighteq$ is a binary relation on a set of alternatives $\mathcal{A}$, that is, $\trianglerighteq \subseteq \mathcal{A} \times \mathcal{A}$. For $(\alpha, \beta) \in \trianglerighteq$, we say "$\alpha$ is preferred to β".

There are many important properties for preference relations (or more generally binary relations) that one can assume. The following is a list of the most common ones. Let $\trianglerighteq$ be a binary relation on $\mathcal{A}$.

Reflexive	For all $\alpha \in \mathcal{A}$, $\alpha \trianglerighteq \alpha$.
Symmetric	For all $\alpha, \beta \in \mathcal{A}$, if $\alpha \trianglerighteq \beta$, then $\beta \trianglerighteq \alpha$.
Transitive	For all $\alpha, \beta, \gamma \in \mathcal{A}$, if $\alpha \trianglerighteq \beta$ and $\beta \trianglerighteq \gamma$, then $\alpha \trianglerighteq \gamma$.
Complete	For all $\alpha, \beta \in \mathcal{A}$, $\alpha \trianglerighteq \beta$ or $\beta \trianglerighteq \alpha$.

We can define some related negations of properties.

Irreflexive	For all $\alpha \in \mathcal{A}$, $\alpha \not\trianglerighteq \alpha$.
Antisymmetric	For all $\alpha, \beta \in \mathcal{A}$, if $\alpha \trianglerighteq \beta$ and $\beta \trianglerighteq \alpha$, then $\alpha = \beta$.
Asymmetric	For all $\alpha, \beta \in \mathcal{A}$, if $\alpha \trianglerighteq \beta$, then $\beta \not\trianglerighteq \alpha$.

Some binary relations with selected properties are commonly known under specific names.

Definition 2.2: Order Relations

A *preorder* is a reflexive and transitive binary relation.
A *partial order* is a reflexive, transitive and antisymmetric binary relation.
A *total preorder* or *weak order* is a complete and transitive binary relation.
A *total order* is a complete, transitive and antisymmetric binary relation.

In the following, we will consider preference relations that are preorders. Associated with a preorder is a symmetric and an asymmetric part. Furthermore, a preorder generates three relations: The corresponding equivalence relation,

strict relation and incomparability relation.

Definition 2.3: Equivalence, Strict and Incomparability Relations

Let $\trianglerighteq$ be a preorder on $\mathcal{A}$.
The symmetric part of $\trianglerighteq$, or *equivalence relation* $\equiv$ is defined by:
For all $\alpha, \beta \in \mathcal{A}$, $\alpha \equiv \beta$, if $\alpha \trianglerighteq \beta$ and $\beta \trianglerighteq \alpha$.

The asymmetric part of $\trianglerighteq$, or *strict relation* $\triangleright$ is defined by:
For all $\alpha, \beta \in \mathcal{A}$, $\alpha \triangleright \beta$, if $\alpha \trianglerighteq \beta$ and $\beta \not\trianglerighteq \alpha$.

The remaining part of $\trianglerighteq$, the *incomparability relation* $\sim$ is defined by:
For all $\alpha, \beta \in \mathcal{A}$, $\alpha \sim \beta$, if $\alpha \not\trianglerighteq \beta$ and $\beta \not\trianglerighteq \alpha$.

2.2.1 Relational vs. Cardinal Preference Representations

We can split preference relations into cardinal and relational preference representations [Wal07].

Cardinal preference representations assign a score or utility to the alternatives, and subsequently alternatives can be compared based on their scores. For numerical utilities, this yields a preference relation that is a total preorder. Utility functions are functions $f : \mathcal{A} \longrightarrow \mathbb{R}$ that assign every alternative a real numbered utility. For alternatives $\alpha, \beta \in \mathcal{A}$, $\alpha \trianglerighteq \beta$ if and only if $f(\alpha) \geq f(\beta)$. Utility functions are well known and studied, and used as preference representations in many papers concerned with decision making [FGE05, Fis70, MD04, WDF+08]. Consider alternatives that are given by feature vectors, so that they are specified over a set of variables $\mathcal{V}$ such that $\mathcal{A} \subseteq \Pi_{X \in \mathcal{V}} \underline{X}$, where $\underline{X}$ is the domain of variable $X \in \mathcal{V}$. In this case, utility functions assume that the variable domains are commensurable and values can be combined into a single score. Moreover, often all variable domains are assumed to be subsets of real or rational numbers. We will discuss some compact representations that rely on additional assumptions on the variables and properties of the function in Section 2.5, however, the results of this dissertation only focus on relational preference representations.

Relational preference representations assume a set of variables $\mathcal{V}$ by which the alternatives are described [BDPP10, BD09, Wal07], i.e., $\mathcal{A} \subseteq \Pi_{X \in \mathcal{V}} \underline{X}$. However, in contrast to utility functions, they do not necessarily assume commensurably

or numerical variable domains. Instead, variables are put into relation with other variables by, e.g., their importance. On the downside, these order relations do not always yield total preorder preference relations.

One famous example of relational orders is given by lexicographic orders. Formally, a lexicographic order is defined as follows.

Definition 2.4: Lexicographic Order

A lexicographic order $\trianglerighteq_{lex}$ of vectors $\mathcal{A}$ with variables $\mathcal{V}$, variable domains $\underline{X}$ with total order relations $\geq_X$ (and strict relation $>_X$), and a total order $X_1 > X_2 > \cdots > X_{|\mathcal{V}|}$ on the variables is given as follows. For alternatives $\alpha, \beta \in \mathcal{A}$, $\alpha \trianglerighteq_{lex} \beta$ if and only if

1. there exists a variable X_i such that $\alpha(X_i) >_{X_i} \beta(X_i)$, and $\alpha(X_j) = \beta(X_j)$ for all $j < i$, or
2. $\alpha(X_i) = \beta(X_i)$ for all $i \in \{1, \ldots, |\mathcal{V}|\}$.

The total order of variables associated with a lexicographic order can be viewed as an importance order of the variables. Alternatives are compared on the most important variable, and only if the values are equal, are the alternatives compared on the second most important variable, and so on. The order relation $\trianglerighteq_{lex}$ is a total preorder on the set of alternatives $\mathcal{A}$. Lexicographic orders find applications in many areas, e.g., multi-objective optimisation problems [Fre04], databases [Ull84], economics [Fis74], mathematics in a broader sense, and many other fields. We discuss lexicographic orders and more generally hierarchical models in more detail for the context of preference representations in Section 2.5.2.

Another example of relational orders is given by Pareto orders, which are defined as follows.

Definition 2.5: Pareto Order

A Pareto order $\trianglerighteq_{Pareto}$ of vectors $\mathcal{A}$ with variables $\mathcal{V}$ and variable domains $\underline{X}$ with total order relations $\geq_X$ is defined by: For alternatives $\alpha, \beta \in \mathcal{A}$, $\alpha \trianglerighteq_{Pareto} \beta$ if and only if $\alpha(X_i) \geq_{X_i} \beta(X_i)$ for all $i \in \{1, \ldots, |\mathcal{V}|\}$.

This order relation allows incomparable pairs of alternatives but is transitive, re-

flexive and antisymmetric. One alternative is only better or equal than another alternative, if it is better or equal in all values of the variables. Thus $\trianglerighteq_{Pareto}$ is a partial order. Pareto orders are applied in voting scenarios [Wal14], multi-objective optimisation problems [BDPP10], database queries [BKS01, MLB15], allocation problems [ACMM05], economics [Sti87, Tia09], and many other fields. We discuss this relational order in more detail in the context of preference representations in Section 2.5.2.

Other relational orders on alternatives could include minimum and maximum operators on the values of variables, or a comparison of value differences where variable domains are numerical [BD09].

2.3 Acquisition of Preference Information

To support users in a decision, we need to know their preferences to understand which options they evaluate as "good" options. That is, we need to have an understating of the user's preference relation by which alternatives are ordered.

Common methods of gaining preference information include the following.

Looking at historical data of the user We can analyse data that explicitly or implicitly indicates user preferences. For example, one can extract preferences over destinations by exploring location aware information obtained from location-based social networks in order to recommend personalised travel packages [YXYG16]. Similarly, tracked browsing behaviour and text mining techniques potentially lead to good recommendations of hotels [LLCH15]. Quite obviously, this requires the existence of historical user data and the consent of the user to use it; both of which might not always be given. In this approach, one also has to identify appropriate time frames in the historical data taken into account, since preferences might change over time.

Considering similar user's behaviour One can assume that, under some measure of similarity, similar users have similar preferences. This assumption is the basis of collaborative filtering methods [SK09]. For example, we can present a user with videos that similar users liked, and expect a high user satisfaction [LRHW17]. Content based recommender systems that employ this method thus require a database of user profiles and a similarity metric [LDGS11]; both

of which might not be realistic in many applications. Furthermore, the resulting recommendations might not be suitable for "grey sheep", i.e., users that do not agree or disagree with any group of people [SK09].

Asking the user directly Although this approach seems to promise very accurate user preference information, it also is the most cumbersome for the user. In contrast to the first two approaches, the users actively spent time and effort to express their preferences. In order to make this approach worthwhile for a user, the time and effort has to match the quality of the recommendations. In the following, we discuss approaches that aim to improve exactly this tradeoff between effort and recommendation quality.

2.3.1 Eliciting Preferences

Bettman et al state that several factors like chance of making an error, justifiability, and the avoidance of conflict influence the user's choice of order relation/preference model when comparing alternatives; however, the (cognitive) effort associated with making a choice is generally assumed to be a major influence on the choice of preference model [BJP90]. The effort of a task can be measured experimentally via completion time of tasks and user self-reported estimation of effort [BJP90], and electroencephalography (EEG) [APGvG10], amongst others. In any elicitation approach that directly involves the user, it is thus crucial to reduce the number and difficulty of questions asked to keep effort and time in an acceptable range for the user.

We need to decrease the number of questions asked while making every possible answer to a question most informative. However, highly expressive preference statements are usually cognitively challenging for a user to express. Thus, an appropriate tradeoff between expressiveness of statements and cognitive effort to express them has to be found. Furthermore, to give the user flexibility in how to express preferences, the chosen approach needs to be as complete as possible, e.g., include negations, conditions, strict and non-strict statements etc. in the underlying preference language. Different approaches of preference elicitation include eliciting relations between outcomes, and eliciting information on the user's preference model (e.g., weights, importance order of variables, etc.).

When assuming an underlying user preference model, even one that the user might not be aware of, it can be beneficial to elicit information on the preference

model rather than direct relations of outcomes. Assume, for example, that the users compare alternatives by a weighted sum of variable values (or utilities). Exact numeric weights, which express the importance or influence of a variable, are very hard or even impossible to formulate for a user [Dav87, KR93]. In fact, although it can be convenient and productive to assume this, the user might not be using, e.g., a weighted sum to compare alternatives. However, by pairwise comparisons of the importance of variables as in [Har06], the relative relation of weights can be elicited. This approach could also be used to elicit a lexicographic or hierarchical model. The number of questions asked in this approach is quadratic in the number of variables. Under other models, like CP-nets, variables are not assumed to be (additively) independent. They can thus not be ordered by importance and the value orders of variables cannot be elicited separately, which usually makes the elicitation process harder [CP04].

An alternative to eliciting information on the user preference model, is to elicit information on the relation between alternatives. It is, for exmaple, easier for a user to express a comparisons between two or more alternatives, than to express numerical weights [DHKP11]. Thus, while comparative statements usually give less information on the user's preference order, this type of preference information can be preferred for elicitation, to avoid high cognitive efforts for users.

Some conversational recommender systems tackle the issue of balancing user effort and recommendation quality by critique based recommendations where a user is repeatedly presented with a set of options [CP12]. This set of options is simultaneously a set of recommended alternatives and a query set to elicit further preferences. The users can identify a preferred alternative under the presented options in order to further improve the recommendations, or decide that they are satisfied with the recommended options. If an alternative α is selected to be preferred amongst a set A of alternatives, this corresponds to a set of comparative preference statements "α is preferred to β" for all $\beta \in A$. This new preference information can be used to compute a new set of "better" recommendations. Once the users decide further improvement on the recommendations is not worth the effort spent on providing preference information, they can terminate the elicitation process. The best balance between effort and quality of recommendation can thus be determined by the user directly and individually.

However, this approach evokes several different issues:

- What properties does an optimal recommendation set need to possess?
- What properties does an optimal query set need to possess?
- How can new preference information be processed to compute the next recommendation/query set?

On the one hand, it is important to define and compute the best query set of alternatives. That is, to find a set of alternatives that gives the most possible information (highest possible reduction of the current set of optimal alternatives) no matter which alternative the user evaluates to be the best. On the other hand, one wants to present the user with the best recommendation set, i.e., a set of solutions that the user is expected to like best. Reasearch for recommender systems investigates the benefit of different properties of the alternatives in the query set, e.g., popularity (based on other users), controversy / diversity, novelty and serendipity, for different applications [SRCP06, MRK06].

Often the optimal query and recommendation set differ and a hybrid solution is found [Bal98]. Some approaches choose to present the user with alternatives of high entropy in order to speed up the elicitation process [SRCP06]. However, in some cases one can prove that the optimal query and recommendation set are the same and no tradeoff has to be found [VB11]. These considerations strongly depend on the assumed type of preference models and the notion of "optimality".

Another query type considered in probabilistic models is the standard gamble, where the users are asked if they prefer an alternative α over a gamble in which the best alternative occurs with probability l and the worst alternative occurs with probability $1 - l$. In probabilistic models the next query to the user is chosen to maximise the expected value (utility) of information [Bou02]. In his partially-observable Markov decision process model, Boutilier incorporates a cost associated with asking a query that reflects both the user's effort for answering the query and the system's effort for computation. He aims at exploring the tradeoff between elicitation effort and decision quality and states that "if the cost of obtaining that information exceeds the benefit it provides, then this information too can be safely ignored" [Bou02].

A more detailed overview of preference elicitation techniques can be found in [CL11] and [CP04].

2.3.2 Learning Preferences

Preference learning tasks, assume that preferences of a user over some alternatives are given, and try to predict the user's preferences over other alternatives. This provides a means of overcoming uncertainty (in the sense of a lack of information) in user preferences. While this approach of acquiring preference information only gives an approximation to the user's preferences, it is able to deal with inconsistent (conflicting or contradicting) input preferences. Many approaches for preference learning use machine learning techniques, and thus require training data to learn their prediction models, and some measure to evaluate their performance. Here, the training data must incorporate labels or comparisons of instances or objects. Active learning algorithms aim at minimizing the labeled data required [Kri07]. They do not assume a full set of labelled training instances. Instead they strategically select instances/objects to be labelled by an expert. Depending on assumptions on the form of preference representation, learning approaches split into learning utility functions and learning preference relations. One of the most prominent problems that preference learning work is concerned with, is learning to rank, which can be categorised in three areas: label ranking, instance ranking, and object ranking [FH10a]. Ranking problems have interesting applications like learning to rank recommendations [BHK98] or search results [Liu09].

Label ranking, seeks to predict a total order on a set of labels for every instance. The training data consists of a set of instances, each associated with a set of observations that give pairwise comparisons of some labels [VG10]. In instance ranking, given some training instances that are labeled and can be ordered according to a given total order on the labels, the goal is to predict a ranking of new instance (possibly by assigning scores or labels) [FH10a]. Object ranker "learn to order things" [CSS98], which are not necessarily represented by features. The input includes some preferences over tuples of objects. The output then consists of a ranking for every set of objects [KKA10].

Consider the learning of utility functions, which assigns a utility to each alternative (by which the alternatives can then be compared). In instance ranking, the input training instances comprise labels already, and the problem is to predict labels for new instances. In contrast, learning methods for label and object ranking deal with constraints on the rankings given by pairwise comparisons of labels or objects. Thus, the utilities of the input instances are not explicitly given. Methods on utility function learning thus vary based on the type of input

data [FH10a].

When learning preference relations, a binary preference relation is sought which extends the ordered tuples of alternatives specified in the input [FH10b]. The difficulty here is to find a preference relation that maximally agrees with the input statements, which can be NP-hard [CSS98].

Similar to our approach to preference inference, model-based preference learning assumes an underlying structure, a preference model, for the wanted preference relation. Learning problems have, for example, been analysed for lexicographic models [BCL+10, YWLdJ10], CP-nets [AD07, GAG13], and general properties on an aggregation operators [Tor10].

Different solutions to ranking problems, depending on their secific tasks and inputs, include the use of regression models [ZHC+04], gradient descent [BSR+05], optimisation approaches [PTA+07], neural networks [Tes89], Bayesian models [HMG07] and others [FH10a].

2.3.3 Inferring Preferences

Because an elicitation process involving the user directly imposes substantial effort on the user, and because learning approaches only deliver an approximation on the user's preference relation, it is only natural to try to make the most of the available preference information.

The *Preference Deduction Problem* aims at deducing new preferences from given ones by using logical deductions. Based on assumptions on the type of preference model and preference language used by the user, new preference statements are deduced with certainty. This problem and the related *Preference Consistency Problem* will be introduced in Section 3.1 and their consideration under different preference languages and qualitative preference models form the main contribution of this dissertation.

Preference inference has previously found applications, e.g., in multi-objective constraint programming in our paper [GRW15] and recommender systems [TWBR11]. It could be employed in any decision problem to handle a lack of preference information.

Papers on deduction and consistency of preferences preceding the work presented in this dissertation studied the problems for CP-nets [DB02, GLTW08],

lexicographic model [Wil14], conditional lexicographic orders [Wil09], sorted-Pareto orders [OW13], linear utility functions [BR07, TWBR11] and preference logics more generally [BLW10].

2.4 Preference Languages

There are different ways in which preferences can be expressed, formalised and interpreted. Generally speaking, preference statements are constraints on the user's preference relation on the considered alternatives. An order relation on the set of alternatives $\mathcal{A}$ is a binary relation $\rhd \subseteq \mathcal{A} \times \mathcal{A}$. Such an order relation satisfies the user's preference statements, if it satisfies all given constraints. Typically, assumptions on the structure of the user's order relation to rank alternatives are made. Transitivity, for example, is a very common and realistic property for orders on alternaives in decision making scenarios [RDDS11, CDS14]. In the remainder of this dissertation we will only consider order relations that are transitive, i.e., if the user prefers A to B and B to C, A is also preferred to C. Different assumptions on the structure of an order relation are discussed in the next section, while this section focuses on constraints given by relations between alternatives. A preference language specifies in which manner these relations between alternatives can be expressed, i.e., which type of statements the user can give. One can consider preference statements as logical formulas [BLW10]. This allows the formulation of conjunctions and disjunctions of preferences as well as any other boolean operator. In the following, however, we present more specific forms of preference statements. In this dissertation, we mainly focus on comparative preference languages, but we also present a general framework to handle unspecified preference languages in Chapter 4. In general, preference languages are divided into quantitative and qualitative, weighted and unweighted preference languages, see p. 4 [Kac11].

When choosing a specific preference language, e.g., for the elicitation of user preferences, one tries to find a tradeoff of how cognitively challenging it is for a user to express a statement and how informative a statement is. A user could for example be asked to provide numeric scores for alternatives, which then would allow one to compare these alternatives to each other, and thus would be quite informative. However, a *quantitative weighted statement* like "I rate a day-time flight with LAN in business class with 6.75 out of 10" can be very challenging for a user to formulate and often leads to imprecise scores. Even

if the user knows properties of the alternatives and is able to compare them by certain criteria, it is not easy for the user to aggregate this information to obtain a numeric score. It would be easier, however less informative, for a user to give a *qualitative weighted statement* like "I like flying with KLM very much". Since no scale is provided, it is unclear *how much* "very much" is especially in comparison with other flight options. However, because these statements are more vague, they naturally are less often wrong.

The intention of expressing preferences is mostly to set different alternatives into a relation, e.g., in order to find the most preferred alternative for a user. Instead of evaluating alternatives individually and comparing them afterwards based on the evaluations, the user could directly give comparisons between alternatives. In this dissertation, the specific preference languages that are considered are comparative languages. We furthermore assume that a set of variables $\mathcal{V}$, by which the alternatives can be compared, is known to the user. For example, flight connections can be compared by the variables airline, time of day and class. The variable *airline* could have a binary domain that contains KLM and LAN. We denote the domain of a variable $X \in \mathcal{V}$ by $Dom(X)$ or $\underline{X}$. To abbreviate the notation, we also write $\underline{Q} = \Pi_{X \in Q} \underline{X}$ for the domain of a set of variables $Q \subseteq \mathcal{V}$. The set of alternatives $\mathcal{A}$ is thus a subset $\mathcal{A} \subseteq \underline{V}$ of complete variable assignments. Sometimes we will refer to alternatives as outcomes.

Ordinal statements of the form "I prefer A to B" are usually easier for a user to express as they allow for a less analytical and more intuitive decision. They can be comparisons of two completely specified alternatives such as "I prefer a night-time flight with KLM in economy class to a day-time flight with LAN in business class", where the only three variables considered are the time of day, class and airline as described before. This statement gives an ordering of two alternatives, which has to be satisfied in the user's preference order on the set of alternatives. Although being easy to formulate, it gives only little information about the whole set of alternatives. We denote such statements by $\alpha \rhd \beta$, where $\alpha, \beta \in \mathcal{A}$ are two alternatives and $\rhd$ is an order relation on the set of alternatives $\mathcal{A}$.

For a user it is cognitively slightly more challenging to express a comparative statement such as "I prefer a flight with KLM to a business class flight with LAN" where only partial information is given on the alternatives involved. The user needs to make a comparison over sets of alternatives, and identify criteria that preferred flights have in common over criteria of less preferred flights.

We denote such statements by $p \rhd q$, where $\rhd$ is an order relation on sets of alternatives, and p and q are partial assignments to variables in $\mathcal{V}$, i.e., $p \in \underline{P}$ and $q \in \underline{Q}$ for variable sets $P, Q \subseteq \mathcal{V}$. Statements like these can imply several constraints on the user's preference order, as they involve sets of alternatives compliant with p and q, and are thus more informative than comparisons of complete assignments of variables. The question of how to interpret this kind of statements arises, i.e., which ordering constraints are implied by such a statement. On the one hand, we have the set L of all KLM flights on the other hand the set R of all business class flights with LAN. The literature discusses different interpretations: ceteris paribus semantics, strong semantics, optimistic semantics, pessimistic semantics and opportunistic semantics, e.g., see [Kac11], that will be outlined in the following.

2.4.1 Semantics

In the following, we will describe different semantics, i.e., understandings of preference statements on the example of the comparative preference statement "I prefer a flight with KLM to a flight with LAN in business class" on partial variable assignments. Let L be the set of all alternatives involving a KLM flight and R the set of of all alternatives involving a business class flights with LAN.

Ceteris Paribus Semantics One of the most common semantics is the "ceteris paribus" (Latin for "all else being equal") semantics, see for example [BBHP99, MD04, RNL+15]. Our example statement will be interpreted as "I prefer a flight with KLM to a flight with LAN in business class given that all other features are equal". We thus only include the tuples in $L \times R$ in the user's preference order in which assignments to all by p and q unspecified variables are equal. This semantics differs from the others by selecting ordering tuples not by ranking within the sets L and R but by specific values of the variables. However, this interpretation is natural in many scenarios [BBD+04a]. We denote order relations $\rhd$ under ceteris paribus semantics by $\rhd_{cp}$.

Strong Semantics In strong semantics, it is assumed that all tuples in $L \times R$ belong to the user's preference order. Thus the previous preference statement is interpreted as: "I prefer any flight with KLM to any flight with LAN in business class". Order relations $\rhd$ under strong semantics are written as $\rhd_{str}$. An

example of a type of preference statement under strong semantics is analysed in [BK01]

Optimistic and Pessimistic Semantics Optimistic semantics interpret the statement in a way that any of the best-ranked alternatives in L has to be preferred to any of the best-ranked alternatives in R. In other words, there has to be an alternative in L that is preferred to all alternatives in R. (This relies on transitivity of order relations, which we assume throughout the whole dissertation, as mentioned earlier.) The statement is thus interpreted as: "There is at least one flight with KLM that is better than any flight with LAN in business class". The pessimistic semantics build the counter part of the optimistic semantics. It is assumed that all worst-ranked alternatives in L are preferred to all worst-ranked alternatives in R, i.e., there has to exist at least one alternative in R that is less preferred that any alternative in L. The statement can thus be formulated as: "There is at least one flight with LAN in business class that is worse than any flight with KLM". We denote order relations $\rhd$ under optimistic and pessimistic semantics by $\rhd_{opt}$ and $\rhd_{pes}$, respectively. [BHK14] shows an example of how to employ optimistic semantics in a multi-objective optimization scenario. They argue that pessimistic and opportunistic semantics could be used in a similar way.

Opportunistic Semantics The opportunistic semantics impose a hybrid between optimistic and pessimistic semantics. We interpret a statement in a way that the best ranked alternatives in L are preferred to the worst ranked alternatives in R. This means at least one alternative in L has to be preferred to at least one alternative in R, i.e., "There exists a flight with KLM that is preferred to some flight with LAN in business class". Order relations $\rhd$ under opportunistic semantics are denoted by $\rhd_{opp}$.
The opportunistic semantics are weaker than the optimistic or pessimistic semantics in the sense that they impose less constraints on the user's preference order on the set of alternatives. Any ordering on a tuple of alternatives that is implied by the opportunistic semantics is also implied by optimistic and pessimistic semantics. Furthermore, any ordering on a tuple that is implied by the ceteris paribus semantics, optimistic semantics, pessimistic semantics or opportunistic semantics is also implied by the strong semantics.

In the following example, we demonstrate the five discussed possible interpre-

tations.

Example 2.1

We consider flight connections which feature the two variables class (business/economy) and airline (KLM/LAN). We consider the four alternatives listed in the following table.

Name	class	airline
α_1	economy	LAN
α_2	economy	KLM
α_3	business	LAN
α_4	business	KLM

The statement φ_1 "I prefer a flight with KLM to a flight with LAN" expresses preference of one set of alternatives $L_1 = \{\alpha_2, \alpha_4\}$ over another set of alternatives $R_1 = \{\alpha_1, \alpha_3\}$. Similarly, the statement φ_2 "I prefer a business class flight to an economy class flight" expresses preference of one set of alternatives $L_2 = \{\alpha_3, \alpha_4\}$ over another set of alternatives $R_2 = \{\alpha_1, \alpha_2\}$. Let the binary relation "preferred to" be denoted by $\rhd$.

In ceteris paribus semantics, the two statements together yield the partial order given by (the transitive closure of) $\alpha_4 \rhd_{cp} \alpha_3 \rhd_{cp} \alpha_1$ and $\alpha_4 \rhd_{cp} \alpha_2 \rhd_{cp} \alpha_1$.

In strong semantics, the two statements together could be inconsistent, i.e., contradicting each other, if the expression "preferred to" is interpreted in a strict way ($\rhd_{str}$ is asymmetric): φ_1 implies that $\alpha_2 \rhd_{str} \alpha_3$, while φ_2 implies that $\alpha_3 \rhd_{str} \alpha_2$. However, if "preferred to" is interpreted in a non-strict way so that equivalences are allowed ($\alpha \equiv_{str} \beta$ if $\alpha \rhd_{str} \beta$ and $\beta \rhd_{str} \alpha$), then the two statements yield the following total preorder: $\alpha_4 \rhd_{str} (\alpha_3 \equiv_{str} \alpha_2) \rhd_{str} \alpha_1$.

In optimistic semantics, under strict order relations φ_1 and φ_2 imply that α_4 is preferred to α_3, α_2 and α_1. However, the order of α_3, α_2 and α_1 is not clear. Similarly, in pessimistic semantics under strict order relations φ_1 and φ_2 imply that α_4, α_3 and α_2 are preferred to α_1. However, the order of α_4, α_3 and α_2 is not clear here. In the case of a non-strict order, we cannot conclude any order of the alternatives for optimistic or pessimistic semantics.

In the opportunistic semantics, no explicit order on tuple of alternatives is

given. However, by φ_1 the user's preference order cannot prefer both α_3 and α_1 to both α_4 and α_2. Similarly, by φ_2 the user's preference order cannot prefer both α_1 and α_2 to both α_4 and α_3.

2.4.2 Types of Comparative Statements

Next, we will outline different important types of comparative statements.

Strict and Non-Strict Preferences In the previous example, we can not only see that the strong semantics implies the most constraints on the user's preference order, but also that it makes a difference whether the statement was interpreted in a strict or in a non-strict way, i.e., if equivalences between alternatives are admitted or not. This is an important classification of comparative preference statements. Allowing indifferences, as in non-strict preferences, and explicitly stating strict preferences are both realistic requirements for the user. We will thus in the following chapters consider languages that allow both types of statements. A non-strict preference statement like "I prefer an economy flight with KLM to a business class flight with LAN" will be formalised as (economy, KLM) $\geq$ (business, LAN). The strict statement "I strictly prefer a business flight with KLM to an economy flight with KLM" is denoted by (business, KLM) $>$ (economy, KLM).

Sometimes, we want to consider the non-strict version $\varphi^{(\geq)}$ of a preference statement φ. This simply means replacing the type of relation between the alternatives in a preference statement by a non-strict relation. If φ is a strict preference statement $\varphi : \alpha > \beta$, for example, then we define $\varphi^{(\geq)}$ as $\alpha \geq \beta$.

Conditional Preferences An important generalisation of the so far discussed preference statements is given by conditional preferences. The user can express that some value order of a variable depends on the values of some other variables. Usually, conditional statements are only considered for unweighted languages [BCL$^+$10, BBD$^+$04a, BEL09, Kac11, Wil04a, Wil11]. However, a weighted statement could also be conditioned. Consider for example the qualitatively weighted statement "I like night-time flights vary much, given that I fly business class". This indeed corresponds to a set of unconditioned qualitatively weighted statements that involve all night-time business class flights.

Conditional unweighted comparative preference statements, like "I prefer a business class flight to a KLM flight, given that the flight is during the night", are restricted to a specific assignment for one or more variables, but might not give detail on preferences involving other assignments for the same variables. It could for example also be the case that KLM flights are preferred to business class flights given that the flight is during the day. Note that unweighted conditional statements can also be expressed by a set of unconditional comparative statements. It is common to denote a statement like "I prefer a business class flight to a KLM flight, given that the flight is during the night" by (night): (business) $\rhd$ (KLM). Depending on the semantics, the same statement can be expressed by (business, night) $\rhd$ (KLM, night). Conditional statements are challenging to handle, however, it is realistic to assume that a user might want to express such conditions. Because of this, some representations of preference orders over alternatives are specifically designed to capture conditionality which are outlined in Section 2.5.2.

Another form of conditionality can be expressed by ceteris paribus statements. In this case, the condition restricts the tuples of ordered alternatives to those with alternatives that have the same (but not a specific) value for some variables.

Negative and Negated Preferences The types of preference statements discussed so far were positive expressions of what a user likes (over something else). It is also possible for the user to express negative preferences that state what they do not tolerate, e.g., "I don't like flying during the night" or "I do not prefer a flight with KLM to a business class flight, given that all other criteria are equal". Bipolar preference languages are languages that include positive and negative preference statements together. Some efforts have been made to find models and operators for bipolar languages [BPRV05, GL10, Kac12].

Similarly, but not equally, we can consider negated statements. A user could express, for example, "It is not true that I prefer flights with KLM to business class flights". This statement under strong semantics could be interpreted as: There exists at least one flight with KLM that is not preferred to some flight in business class. Note that in contrast, the negative statement "I do not prefer flights with KLM to business class flights", under strong semantics, states that all KLM flights are not preferred to business class flights. In this dissertation, we lay a focus on unweighted ordinal comparative preference statements and

in this context also consider negated statements. While negative statements are not considered in the specified preference languages in this dissertation, Chapter 4 provides a general framework for unspecified preference languages, which thus may include negated or negative preferences.

If the assumed preference order can be incomplete, the negation of a statement cannot necessarily be expressed by a positive statement. However, if the preference order is complete, then negated statements can be substituted by different positive statements. Assume in the following that the preference order $\rhd$ on the alternatives $\mathcal{A}$ is complete, i.e., for all alternatives $\alpha, \beta \in \mathcal{A}$ either $\alpha \rhd \beta$ or $\beta \rhd \alpha$. The negation of a non-strict comparative statement $\alpha \geq \beta$ on complete alternatives $\alpha, \beta \in \mathcal{A}$ is simply the strict preference statement $\beta > \alpha$, since $\neg(\alpha \geq \beta)$ expresses that α is not preferred or equal to β, and thus, by completeness of the order, β is strictly preferred to α. Similarly, the negation of a strict statement $\alpha > \beta$ is the non-strict preference statement $\beta \geq \alpha$.

For comparative statements on partial assignments of the variables, the negation of a statement depends on the chosen semantics. For example, under strong semantics the negated statement $\neg(p \geq_{str} q)$ expresses that there exist alternatives α and β that extend p and q, respectively, such that $\beta > \alpha$. This corresponds to the opportunistic statement $q >_{opp} p$. Similarly, negated strict statements under strong semantics can be expressed by non-strict statements under opportunistic semantics. Hence, it is also true that the statement $\neg(p \geq_{opp} q)$ is equivalent to $q >_{str} p$, and $\neg(p >_{opp} q)$ is equivalent to $q \geq_{str} p$.

The statement $\neg(p \geq_{opt} q)$ under optimistic semantics expresses that there exists an alternative that extends q that is strictly preferred to the most preferred alternative that extends p and is thus strictly preferred to all alternatives that extend p. Hence, the statement $\neg(p \geq_{opt} q)$ is equivalent to $q >_{opt} p$. Thus, $\neg(p >_{opt} q)$ is equivalent to $q \geq_{opt} p$. For pessimistic semantics, we can show by a similar analysis that $\neg(p \geq_{pes} q)$ is equivalent to $q >_{pes} p$ and $\neg(p >_{pes} q)$ is equivalent to $q \geq_{pes} p$.

2.5 Compact Representations of Preference Relations

Often constraints on the structure of the user's preference relation on the alternatives are imposed in order to enable a simpler or more efficient analysis. We call the structural representation of an order relation a *preference model*. One example is given by lexicographic orders, for which the required structure is given by a total order on the variables. In the following, we present preference models and typical properties of preference relations that can be demanded depending on the application.

As argued before, it is reasonable to demand preferences to be *transitive*, i.e., if $\alpha \rhd \beta$ and $\beta \rhd \gamma$ for alternatives $\alpha, \beta, \gamma \in \mathcal{A}$ and order relation $\rhd$ on the alternatives then also $\alpha \rhd \gamma$ holds. One example of a preference model, in which some form of non-transitivity is considered, is given when alternatives are represented as intervals, e.g., a time interval, and one alternative is preferred to another if its interval lies completely to the right of the other alternatives interval [Fis85, ÖT06]. Here, the incompatibility of alternatives is not transitive. All preference models presented in the following induce transitive order relations on the alternatives.

An order relation $\rhd$ is called *antisymmetric* when for any two alternatives $\alpha, \beta \in \mathcal{A}$ with $\alpha \rhd \beta$ and $\alpha \neq \beta$, we have $\beta \not\rhd \alpha$. One example of an antisymmetric order relation is the strict order discussed in the previous section, in which it is not possible that two alternatives are equally preferred.

A common differentiation is made in the completeness of order relations. If an order relation $\rhd$ on the alternatives satisfies $\alpha \rhd \beta$ or $\beta \rhd \alpha$ for all pairs of alternatives $\alpha, \beta \in \mathcal{A}$, then $\rhd$ is *complete* (see Section 2.2). For example, total orders or total preorders are complete order relations, whereas a partial order is not complete.

A partial order induces an *incomparability relation* $\sim$ on the alternatives, i.e., $\alpha \sim \beta$ if and only if $\alpha \not\rhd \beta$ and $\beta \not\rhd \alpha$ for $\alpha, \beta \in \mathcal{A}$. In some applications it might make sense that the user could express that he cannot make a statement about the relation of two alternatives. For example a user could express that he finds the film "Titanic" neither better nor worse nor equally good as the film "Scary Movie" as they are two completely different genre, and thus incomparable. This concept of incomparability manifests itself in the English language as the idiom

of "comparing apples and oranges". An important example of an order relation that is a partial order is a Pareto order. We will discuss Pareto models, preference models based on Pareto orders, in detail later in this dissertation (Section 2.5.2, 3.3, 4.3.3 and Chapter 5).

When the variable values of alternatives are aggregated into one numerical score by which the alternatives can be compared, all properties of the natural relation $\geq$ on the rational numbers hold, e.g., completeness, transitivity and reflexivity.

From the viewpoint of voting theory, the alternatives can be seen as candidates, variables correspond to voters and the value orders for variables are used to imply a preferences relation over the candidates. A number of properties of the preference relation over alternatives are natural to consider to establish fair or democratic votes, e.g., non-dictatorship, anonymity, weak Paretianity, and independence of irrelevant alternatives (see details under Social Choice Functions in Section 2.5.1).

The user is typically assumed to express her or his preferences by a preference model, which means that the user's order relation on alternatives can be expressed compactly. Thus, instead of listing all tuples in the order relation (which can be exponentially many depending on the number of alternatives in $\mathcal{A}$), we can state rules on the variables and their value domains by which the order of alternatives can be determined.

Compact representations often allow to handle preferences more efficiently, but also reflect the nature of the alternatives in the eyes of the user. Variables can be independent, or the value orders of some variables might depend on the assignment of other variables. All variables can be considered as equally important, or assigned different importance values / levels. The domains of variables can be non-commensurable, or variable values can be combined so that they allow tradeoffs. Some preference models imply complete orders, whereas others allow incomparable pairs. Furthermore, the implied order relations on the alternatives have different properties, such as symmetric, antisymmetric, transitive or intransitive.

More differentiations arise when we consider preference models in connection with given preference statements that have to be satisfied. Possiblistic logic, for example, can handle inconsistent preferences because it provide a measure for the degree of inconsistency [DLP94]. Many other models discussed

in this dissertation, cannot express inconsistent preferences. Some models are restricted to particular preference languages. Not every preference model can satisfy conditional preference statements for example. Properties, algorithms and complexity results that are connected to specific languages and problems are discussed separately in the following chapters for some of the presented preference models.

In the remainder of this section, we give an overview of popular and important types of preference models. We point out further properties, advantages and disadvantages in our description.

2.5.1 Aggregation Functions

When alternatives have multiple features, i.e., are represented as vectors on several variables, then one idea to compare alternatives is to aggregate the values assigned to the variables. We discuss the most common aggregation functions that have been developed in economics, operations research and social choice. A broader presentation of these can, for example, be found in Chapter 17 in [BDPP10]. The general idea is to assign a numerical value to every alternative so that alternatives can be compared on a global scale. This automatically provides a ranking of all alternatives, which enables us to find optimal alternatives or determine the order of two alternatives. However, one drawback is that variable domains need to be commensurable.

The introduced aggregation functions are developed for complete assignments of variables and commensurable variables which typically have domains in $\mathbb{Q}$ or $\mathbb{R}$. We will assume that the variable domains for variables $X \in \mathcal{V}$ are given by the real numbers $\underline{X} = [0, 1]$. For better readability, we denote the assignment $\alpha(X_i)$ of an alternative $\alpha \in \mathcal{A}$ to a variable $X_i \in \mathcal{V}$ by α_i. Thus every alternative α can be represented by a tuple $(\alpha_1, \ldots, \alpha_n)$, where n is the number of variables $\mathcal{V}$.

Naturally, aggregation functions imply total preorders on alternatives, as they assign every alternative a single value which is comparable to every other alternative. This aggregated value however, can only be computed if complete information on an alternative is given. This means, aggregation functions only allow to compare complete variable assignments, and not partial assignments to variables. There are different other properties that characterise aggregation

functions, such as independence, associativity, commutativity of variables and stability under linear transformation.

Averaging Operators

There are several general definitions of operators that are able to express averages, minimum and maximum. In the averaging operators described next, variables are implicitly assumed to be independent, i.e., their values do not depend on other variables assignments.

The general definition of the quasi-arithmetic mean of an alternative $\alpha \in \mathcal{A}$ is given by $M_f(\alpha) = f^{-1}(\frac{1}{n}\sum_{i=1,\dots,n} f(\alpha_i))$, where f is a continuous strictly monotonic function. When f simply is the identity, then M_f is the usual arithmetic mean. Similarly, the quadratic, geometric, harmonic, root-mean-power and exponential mean can be constructed by choosing appropriate functions f, see pp. 684 in [BDPP10]. We can consider a weighted version of quasi-arithmetic means, defined as $M_{f,w}(\alpha) = f^{-1}(\sum_{i=1,\dots,n} w_i f(\alpha_i))$, which gives every variable an importance factor w_i. As before, f is assumed to be a continuous strictly monotonic function.

Both variants are not stable under linear transformation, i.e., for operator $\Omega = M_f$ or $\Omega = M_{f,w}$ and numbers $c, r \in \mathbb{R}$, $\Omega(c * \alpha_1 + r, \dots, c * \alpha_n + r)$ is not necessarily the same as $c * \Omega(\alpha_1, \dots, \alpha_n) + r$. However, Aczél [Acz48] proves that in the unweighted case f, is determined up to a linear transformation, i.e., $M_f = M_{rf+s}$ for $r, s \in \mathbb{R}$ and $r \neq 0$. While the unweighted quasi-arithmetic mean is commutative, i.e., variable positions could be swapped, the weighted version is not since weights are specifically assigned to particular variables.

The common median operator is commutative and invariant under linear transformation. A weighted version of the median operator can be constructed by extending the variable set by duplicates. Other works explicitly aim to consider simple averaging operators in the form of weighted sums for preference inference problems, which are invariant under linear transformation [WM16].

An ordered weighted averaging operator OWA_w with weights $w = (w_1, \dots, w_n)$ is defined on an alternative $\alpha = (\alpha_1, \dots, \alpha_n)$ as the weighted average of the ordered sequence $\alpha_{(1)}, \dots, \alpha_{(n)}$, where $\alpha_{(1)} \leq \dots \leq \alpha_{(n)}$ [Yag88]. Similar to the median operator, OWA_w is commutative and invariant under linear transformation because it considers the positions of values in an ordered se-

quence. Formally we define $OWA_w(\alpha) = \sum_{i=1,\dots,n} w_i \alpha_{(i)}$. Note that OWA_w can represent the arithmetic mean (by setting $w = (1/n, \dots, 1/n)$), the unweighted minimum (by setting $w = (1, 0, \dots, 0)$) and the maximum (by setting $w = (0, \dots, 0, 1)$). In [DP86], a weighted minimum and weighted maximum operator are introduced as $\min_w(\alpha) = \max_{i=1,\dots,n} \min(1 - w_i, \alpha_{(i)})$ and $\max_w(\alpha) = \min_{i=1,\dots,n} \max(w_i, \alpha_{(i)})$, respectively, where $\max_{i=1,\dots,n} \alpha_i = 1$. These weighted minimum and maximum operators are commutative and invariant under linear transformation.

To overcome the independence-requirement of variables, one can consider (discrete) Choquet integrals and (discrete) Sugeno integrals, which are well studied in the field of multi-criteria decision making [Cho54, DMP+01, Gra96, Gra03, GL02, GL10, GR00, LG03, Sch86, Sug74]. The discrete Choquet integral is defined similarly to ordered weighted averaging operators, except that the weights depend on the fuzzy measure of subsets of the ordered sequence of variable values. A *fuzzy measure* on the index set $N = \{1, \dots, n\}$ is a function $\mu : 2^N \longrightarrow [0, 1]$ with $\mu(\emptyset) = 0$ and $\mu(N) = 1$ and $\mu(A) \leq \mu(B)$ for $A \subseteq B$ (μ is monotonic). Formally, the discrete Choquet integral with respect to some fuzzy measure μ is defined as $C_\mu(\alpha) = \sum_{i=1,\dots,n} \alpha_{(i)}[\mu(\{\alpha_{(i)}, \dots, \alpha_{(n)}\}) - \mu(\{\alpha_{(i+1)}, \dots, \alpha_{(n)}\})]$, where $\alpha_{(1)}, \dots, \alpha_{(n)}$ is the ordered sequence for alternative $\alpha = (\alpha_1, \dots, \alpha_n)$, meaning $\alpha_{(1)} \leq \dots \leq \alpha_{(n)}$. The discrete Sugeno integral uses a fuzzy measure as weights as well and resembles a weighted maximum operator on the ordered sequence of variable values. Formally, the discrete Sugeno integral with respect to some fuzzy measure μ is defined as $S_\mu(\alpha) = \max_{i=1,\dots,n} \min(\alpha_{(i)}, \mu(\{\alpha_{(i)}, \dots, \alpha_{(n)}\}))$.

Utility Functions

Among the most common preference models are utility functions. A function $f : \mathcal{A} \longrightarrow \mathbb{R}$ is used to determine the utility, a numeric value, of every alternative. Generally, a utility function need not have a compact presentation. In the following we present cases in which compact presentations are possible.

The maybe simplest form of a utility function is a weighted sum, which belongs to the class of the weighted quasi-arithmetic means presented in the last subsection (where f is the identity). In this case a preference model can be represented compactly by a weight vector $w \in \mathbb{R}^n$. In weighted sum (or weighted average) models variables are considered to be independent, i.e., variable values do not influence other variable's value orders. A set of variables $Y \subseteq \mathcal{V}$ is

preferentially independent of its complement $\mathcal{V} \setminus Y$ (for some binary relation $\succcurlyeq$ on the alternatives), if for any $y, y' \in \underline{Y}$ and $z, z' \in \underline{\mathcal{V} \setminus Y}$, we have $(y, z) \succcurlyeq (y', z)$ implies $(y, z') \succcurlyeq (y', z')$.

More generally, if variables are mutually preferential independent, i.e., any set of variables $Y \subseteq \mathcal{V}$ is preferentially independent of its complement $\mathcal{V} \setminus Y$, then we can write the utility function f as a sum of utilities $f(X_1, \ldots, X_n) = m(\sum_{i=1,\ldots,n} u_i(X_i))$, where m is a monotone function and $n \geq 3$ the number of variables [AS15]. We call such a utility function *additive independent*.

A more general definition of utility functions allows dependences between variables. A variable cover with *factors* $Z_1, \ldots, Z_k \subseteq \mathcal{V}$ and $\mathcal{V} = \bigcup_{i=1,\ldots,k} Z_i$ is defined to be *generalised additive independent (GAI)* for utility function f, if there exist utility functions $u_i : Z_i \longrightarrow \mathbb{R}$ such that $f(X_1, \ldots, X_n) = \sum_{i=1,\ldots,k} u_i(Z_i)$. Reversely, f is a *generalised additive independent* utility function over $Z_1, \ldots, Z_k$, if it can be additively decomposed by factors $Z_1, \ldots, Z_k$. A GAI-net is a graph with nodes representing all factors $Z_1, \ldots, Z_k$ for utility function f. In case the intersection of factors Z_i and Z_j is non-empty, the graph contains an edge $\{Z_i, Z_j\}$ that is labelled by $Z_i \cap Z_j$. Thus, the edges reflect the dependency between factors and independent factors are unconnected nodes. Because of their acknowledgement of possible dependencies between variables, GAI-nets have been studied in decision making for efficient ranking and recommendation of alternatives with multiple attributes [DGP09a, DGP09b, GPQ06].

Social Choice Functions

Social choice theory and more specifically voting theory considers group decision making scenarios, where every voter is represented by a preference relation over candidates. In our previous notation, the voters correspond to variables and the candidates correspond to alternatives. Note that this implies that the complete set of alternatives is known and that the variable domains are ranks from 1 to the number of candidates. The preferences of all voters are then aggregated to find one or several winner candidates. These aggregation methods have been analysed and evaluated for many different criteria most of which are specifically relevant for the application of democratic elections.

It is, for example, important in a democratic election that all voter's preferences are treated as equally important, i.e., that the voters are anonymous. This is in direct contrast to any kind of weighted or hierarchical models in which each

variable is of different importance. When one considers variables that represent features like costs, distances, etc. instead of voters, it might be natural that some features are more decisive than others. Similarly, non-dictatorship and neutrality are often desired. Non-dictatorship means that there cannot exist a voter that decides the vote independently of the other voter's preferences. Neutrality expresses that all candidates are treated equally, i.e., in case of a tie between candidates no candidate is preferred to another because of external criteria like skills, gender, seniority, etc. Again, this applies specifically to voting scenarios. For example, when deciding between cars, in case of a tie one specific color might be preferred (although the color is not crucial to the decision and therefore not represented as a variable).

Other properties considered in this field can also be transferred to different decision making scenarios. Unanimity, for example, requires that if all preference relations given by the voters (variables) prefer one outcome over another, then the aggregated preference relation should agree with this. This corresponds to strict Pareto dominance. If one alternative is preferred to another in all criteria, the it should be overall preferred. Universality ensures that any ranking of alternatives is possible in the voter's preferences. Independence of irrelevant alternatives is given, if for any two alternatives, their overall relative ranking only depends on their relative rankings within the voters' preferences, and not on the ranking of other alternatives.

One of the most important results in voting theory is Arrow's proof of the impossibility to have an aggregation function that induces a total preorder on the alternatives, for three or more candidates and a finite number of voters and simultaneously satisfies: universality, independence of irrelevant alternatives, unanimity and non-dictatorship [Arr50]. This and many other imposibillity results (e.g., Gibbard-Satterthwaite's theorem [Sat75, Gib73], Sen's theorem of the paretian liberal [Sen70]) show the need of finding compromises between properties of the aggregation methods. However, as indicated before, the impossibility results are not necessarily relevant to other decision making problems.

As examples of social choice functions, we consider two of the most studied ones: the Condorcet and the Borda Count. Under the Condorcet method, an alternative (or candidate) is preferred to another, following the majority rule, if it is preferred in more variables than the other. This means, for alternatives $\alpha, \beta \in \mathcal{A}$, $\alpha \geq_{Condorcet} \beta$ if $|\{X_i \in \mathcal{V} \mid \alpha_i > \beta_i\}| \geq |\{X_i \in \mathcal{V} \mid \beta_i > \alpha_i\}|$. As

expected in a democratic voting scenario, under this order relation variables (voters) are equally important and in particular anonymous [BMP09]. Moreover, a sort of tradeoff between votes is considered; one voter's preference can be redeemed by another voter's reverse preference. The resulting order relation is known to be respecting unanimity (meaning if one alternative is better than another in all variables, then it is preferred to the other), independent of exact ranks (meaning only the relation between variable assignments and not the exact variable values are important) and non-dictatorial, see e.g., [BMP09]. Furthermore, the resulting order on the alternatives is an important example of a non-transitive order, which nonetheless seems to be a relevant voting rule [Fis77]. One consequence of the non-transitivity is that for a set of alternatives and value orders on variable domains, there does not always exist a winner, i.e., an alternative that is preferred to all other alternatives. This phenomena is known as the Condorcet Paradox [BCE+16]. Generally, when considering a decision scenario for a single user, we do not aim to determine a single alternative that is preferred to all others. Rather, we search for a presentable set of good options to present to the user. However, in a democratic election, the existence of a winner can be an inevitable necessity.

By the Borda count, given that we have perfect knowledge of all alternatives and their variable assignments, we can immediately select an overall winner, by comparing the sum of ranks of alternatives within the variable domain orders. The rank of one alternative α for one variable X is computed by the position of the value $\alpha(X)$ in the total value order $\geq_X$ on the variable domain $\underline{X}$. More specifically, $rk_X(\alpha) := |\{d \in \underline{X} \mid d > \alpha(X)\}|$. Hence, for alternatives $\alpha, \beta \in \mathcal{A}$, $\alpha \geq_{Borda} \beta$ if $\sum_{X \in \mathcal{V}} rk_X(\alpha) \leq \sum_{X \in \mathcal{V}} rk_X(\beta)$. This order relation on alternatives, as opposed to the Condorcet method, always guarantees a winner. Interestingly, if a Condorcet-winner exists, it is not necessarily chosen as a winner by the Borda count. Among other properties, a voting based on the Borda count is non-dictatorial, anonymous, transitive, respecting unanimity, and neutral with respect to the order of voters [BMP09].

2.5.2 Qualitative Preference Models

Qualitative models describe a relation between variables $\mathcal{V}$ by which the alternatives are described. In contrast to aggregation functions, they do not assign a numerical value to alternatives and thus do not necessarily assume commen-

surably or numerical variable domains. In the following we describe important examples.

Pareto Models

Pareto orders give a natural way of comparing alternatives; one alternative is preferred to another if it is better in all (relevant) variables. This order relation is widely studied in many fields concerning preferences, sometimes under the terms of unanimity, Pareto efficiency or Pareto optimality. In voting scenarios, the concept of unanimity [Wal14] makes for complicated decisions, since all voters (variables) have to agree upon preferring one alternative to another. Reversely, if the goal is to rule out alternatives that are worse than others, Pareto dominance makes for a very cautious rule. Only alternatives, which are worse in at least one variable than another alternative and equal in all remaining variables, i.e., Pareto dominated by another alternative, are discarded. A Pareto frontier is a set of alternatives, which are not Pareto dominated by any other alternative. These sets are sometimes computed as "optimal solutions" to multi-objective optimisation problems [BDPP10]. In database queries the skyline operator reflects the same principle [BKS01, MLB15]. Other application can be found in allocation problems (Pareto optimality) [ACMM05] and more generally in economics (Pareto efficiency) [Sti87, Tia09].

The simplest form of a Pareto model is a set $P = \{\{X_1\}, \ldots, \{X_k\}\}$ of singleton variable sets. An alternative $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succcurlyeq_P \beta$, if and only if $\alpha(X) \geq_X \beta(X)$ for all variables X in P. $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$, written $\alpha \succ_P \beta$, if and only if $\alpha \succcurlyeq_P \beta$ and there exists a variable X in P such that $\alpha(X) >_X \beta(X)$.

This can be interpreted as unanimity on features. For example, based on Pareto orders, a flight connection A is preferred to a flight connection B, if A is better in all aspects: the class, the travel time and the airline. The involved variables are thus equally important and there is no compromise or tradeoff between different variables possible. We will discuss these models in more detail in Section 3.3 and outline important generalisations.

Variable Hierarchies

Lexicographic orders are widely known and often used as order relations on tuples. The order corresponds to the way words/articles are sorted alphabetically in a lexicon or dictionary. The order of two words is determined by considering the first letter of the words, only if these are equal, the second letter is considered, and so on. As preference models they assume a hierarchical structure on the variables. This can be realistic in many scenarios. A user could for example decide to take the cheapest flight connection possible to go from New York to Rome. Only if flights are equally cheap, the user might consider the travel time and prefer a day-time flight to a night-time flight. Although Keeney and Raiffa [KR93] believe that the use of lexicographic orders is "rarely appropriate", Bettman et al.'s study in [BJP90] suggests that a comparison of alternatives by a lexicographic order is relatively effortless and time efficient for a user. Due to the intuitive nature of this order relation, it is no surprise that many works consider lexicographic preference models, see [BCL$^+$10, BH12, DIV07, Fis74, Fis75, FM07, FHWW10, Kno00, KJ07, Wil14, YWLdJ10] and our works [GRW15, WG17].

A fvo lexicographic model L is given by an ordered tuple $(X_1, \ldots, X_l)$ of a subset of all-different variables $\{X_1, \ldots, X_l\} \subseteq \mathcal{V}$. The set of fvo lexicographic models is denoted by $\mathcal{H}(1)$. Assuming that for every variable $X \in \mathcal{V}$ there exists a fixed associated total value order $\geq_X$ on the domain, we can compare alternatives by such a tuple of variables in the usual lexicographic way (see Definition 2.4: $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under L, written $\alpha \succ_L \beta$, if and only if there exists $1 \leq i \leq l$ with $\alpha(X_i) >_{X_i} \beta(X_i)$ and for all $j < i$, $\alpha(X_j) = \beta(X_j)$. $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under L, written $\alpha \succcurlyeq_L \beta$, if and only if $\alpha \succ_L \beta$ or $\alpha(X_i) = \beta(X_i)$ for all $1 \leq i \leq l$. Given a fvo lexicographic model $L \in \mathcal{H}(1)$, we compare alternatives at the first, most important, variable; only if the value assignments are equal, the second most important variable is considered, and so on. Thus a fvo lexicographic model L represents a strict importance order on the variables involved.

However, this condition can be relaxed by considering hierarchical models which allow ties between the importances of variables. Preference models with variable hierarchies as we defined them in [WGO15] are called HCLP models and borrow their name from Hierarchical Constraint Linear Programming [WB93], where feasible solutions are compared by a kind of generalised lexicographic order. They combine values of variables in the same importance

level by an operator. Instead of considering an ordered tuple of variables as in fvo lexicographic models, we consider an ordered tuple of sets of variables. Thus, instead of considering a strict order on variables, hierarchical models imply a total preorder on a subset of variables that can be represented by *level sets* of equally important variables. This allows tradeoffs between variables of the same level set.

In addition to the just described variable hierarchies, we can also consider a generalised form of lexicographic model. A cvo lexicographic model again consists of a total order on a subset of variables. Additionally, variables in this tuple are annotated with a value order. This follows the assumption that value orders on variable domains are not fixed and specified in the preference model.

The mentioned lexicographic and hierarchical models are discussed in more detail in Section 3.4.

Conditional Preference Networks

A qualitative counterpart to GAI-nets, which describe conditional preference relations in a quantitative way, is given by CP-nets. This type of preference model and several variations have been extensively studied for the last twenty years, e.g., see [AGJ+17, BBD+04a, BBD+04b, BBHP99, BD02, BD04, DRVW03, GLTW08, LVK10, PRVW04, Wil04b].

In CP-nets variables can be dependent, i.e., the value order of one variable can depend on the value assignments of other variables. Let $\succeq$ be the preference relation on outcomes given by a CP-net. Then a set of variables $Y \subseteq \mathcal{V}$ is *preferentially independent* (from its complement) if for all $y, y' \in \underline{Y}$ and $z, z' \in \underline{V \setminus Y}$, $yz \succeq y'z$ if and only if $yz' \succeq y'z'$ (which is if and only if $y \rhd_{cp} y'$ under ceteris paribus semantics). Similarly, for condition $T \subseteq V \setminus Y$, set of variables $Y \subseteq \mathcal{V}$ is *preferentially independent under condition* T if for all $y, y' \in \underline{Y}$ and $z, z' \in \underline{(V \setminus Y) \setminus T}$, $yz \succeq y'z \mid T$ if and only if $yz' \succeq y'z' \mid T$ (which is if and only if $y \rhd_{cp} y' \mid T$ under ceteris paribus semantics). Here, the notation $yz \succeq y'z \mid T$ means that any alternative α extending yz is preferred to any alternative β extending $y'z$, given that α and β have the same values for variables T. This allows CP-nets to handle conditional preference statements.

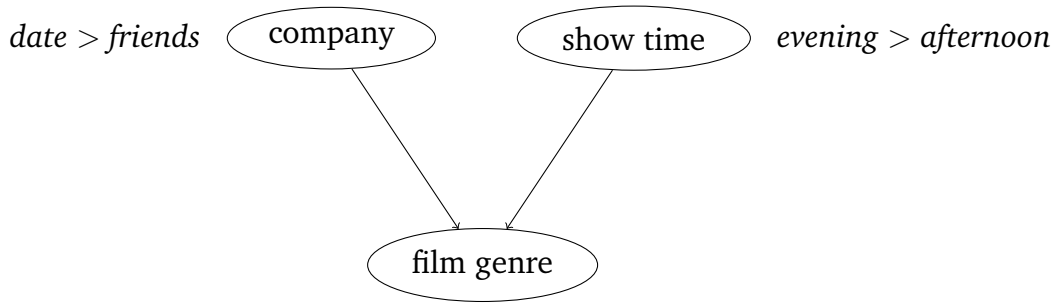
As for GAI-nets, we can represent CP-nets in a graphical way. Here, the nodes represent variables. Directed edges are given by the dependencies of variables,

i.e., the directed edge (X, Y) represents that the value assignments for the variable associated with X influencing the value order of the variable associated with Y. Every node is annotated with a conditional preference table that captures the conditions on the associated variable given by all its predecessors, given a ceteris paribus semantics that keeps all successor variables equal.

Example 2.2

Consider the choice between different films in the cinema. Suppose that the variable company taking values (friends/date) influences the preference over the show time (afternoon/evening), and both influence the preference over the film genre (horror/drama/comedy).

Let the complete preference structure be given by the following CP-net.

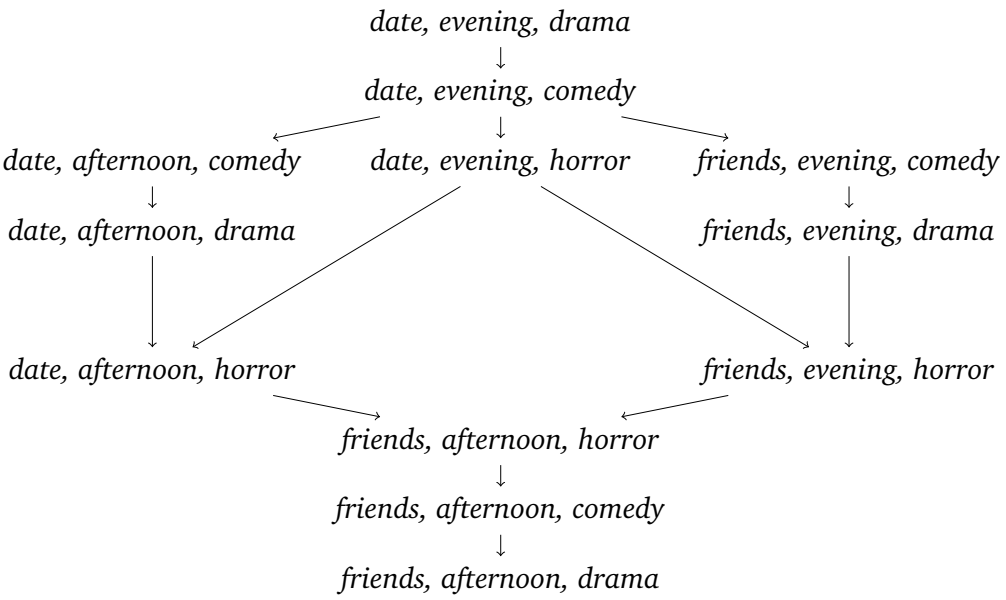


date, evening:	*drama* $>$ *comedy* $>$ *horror*
date, afternoon:	*comedy* $>$ *drama* $>$ *horror*
friends, afternoon:	*comedy* $>$ *drama* $>$ *horror*
friends, evening:	*horror* $>$ *comedy* $>$ *drama*

The conditional preference tables induce preferences by ceteris paribus semantics that are given in the following table.

Preference Statement	Local Preferences
date > *friends*	*date, evening, drama* > *friends, evening, drama* *date, evening, comedy* > *friends, evening, comedy* *date, evening, horror* > *friends, evening, horror* *date, afternoon, drama* > *friends, afternoon, drama* *date, afternoon, comedy* > *friends, afternoon, comedy* *date, afternoon, horror* > *friends, afternoon, horror*
evening > *afternoon*	*date, evening, drama* > *date, afternoon, drama* *date, evening, comedy* > *date, afternoon, comedy* *date, evening, horror* > *date, afternoon, horror* *friends, evening, drama* > *friends, afternoon, drama* *friends, evening, comedy* > *friends, afternoon, comedy* *friends, evening, horror* > *friends, afternoon, horror*
date, evening: *drama* > *comedy* > *horror*	*date, evening, drama* > *date, evening, comedy* *date, evening, comedy* > *date, evening, horror*
date, afternoon: *comedy* > *drama* > *horror*	*date, afternoon, comedy* > *date, afternoon, drama* *date, afternoon, drama* > *date, afternoon, horror*
friends, evening: *comedy* > *drama* > *horror*	*friends, evening, comedy* > *friends, evening, drama* *friends, evening, drama* > *friends, evening, horror*
friends, afternoon: *horror* > *comedy* > *drama*	*friends, afternoon, horror* > *friends, afternoon, comedy* *friends, afternoon, comedy* > *friends, afternoon, drama*

In general, the order relation given by a CP-net is a strict partial order (i.e., a transitive irreflexive order relation) [Wil04b]. The resulting partial order on the set of alternatives (all possible assignments) in our example is given below. Here, $\alpha \to \beta$ expresses that alternative α is preferred to alternative β.



2.6 Summary

Preferences find many applications in the area of artificial intelligence. When handling user preferences one has to take various factors into account: user effort, expressiveness, information value, complexity of representation, computational complexity. Many combinations of preference languages and preference models to represent preference relations have been explored in the literature to handle user preferences. They all carry different properties that can be justified depending on their application. However, natural additions to the literature arise when considering qualitative models. In this dissertation, we focus on defining novel preference model types and analysing their properties with respect to the Consistency and Deduction Problem. Lexicographic models of sorts have been considered in related work to some extent. In this dissertation, we extend these promising results to more general lexicographic orders and more complex preference languages. Exploiting the idea of basing a type of preference model on a common type of qualitative order relation like for lexicographic models, we design Pareto models which are based on Pareto orders. These are to the best of our knowledge also novel preference models. Furthermore, we incorporate the idea of allowing tradeoffs between features by the design of interesting new semi qualitative preference models: hierarchical models and general Pareto models.

Chapter 3

Preliminaries

In this dissertation we analyse the Preference Deduction Problem and the Preference Consistency Problem for some qualitative preference models that are based on lexicographic and Pareto orders and for different languages of comparative strict and non-strict preference statements on complete and partial variable assignments. We introduce the considered problems in the first section of this chapter and then give detailed descriptions and relevant definitions of the preference languages and models considered in the remainder of the dissertation.

3.1 Preference Inference

Because an elicitation process involving the user directly imposes substantial effort on the user, it is only natural to try to make the most of the elicited information. Preference inference aims at deducing new preferences from given ones by using logical deductions. Based on assumptions on the type of preference model and preference language used by the user, new preference statements are deduced with certainty (assuming the correctness of the assumptions).

The Preference Deduction Problem asks, if another preference can be deduced from a set of given user preferences with "certainty", whereas preference learning techniques try to find "likely" deductions. We will give a closer explanation to this approach of preference inference in the following.

Consider a user that needs to make a choice over a set of alternatives $\mathcal{A}$. Let $\Gamma = \{\varphi_1, \ldots, \varphi_k\}$ be a set of preference statements φ_i for $i = 1, \ldots, k$, which

the user regards to be true. In order to help the user decide, we need to find out which the most preferred alternatives in $\mathcal{A}$ are. We can do so, by analysing which other preferences can be deduced. Then, the user can be presented with all alternatives that are not "with certainty" less preferred than another alternative. For these deductions we make certain (reasonable) assumptions on the preferences of the user.

Assume the user models preferences in a certain way. For example, the user could always prefer alternatives in a certain price range, or the user could compare alternatives by a weighted sum of certain criteria. We can then define a set $\mathcal{G}$ of *preference models* that are representations of order relations on the set of alternatives, which are compliant with the way the user is assumed to model preferences. Following the previous examples, the set $\mathcal{G}$ could consist of different price ranges that might be acceptable for the user, or $\mathcal{G}$ could include different weight vectors for the weighted sum of certain criteria the user might use.

Based on a set of preference statements $\mathcal{L}$ and models $\mathcal{G}$ we can define a satisfaction relation $\models$. If a preference model $\pi \in \mathcal{G}$ agrees with, or *satisfies*, a preference statement $\varphi \in \mathcal{L}$, we write $\pi \models \varphi$. This expression means that the order relation on alternatives associated with the preference model π satisfies all constraints that are specified by the preference statement φ. If, for example, the statement φ expresses that an alternative $\alpha \in \mathcal{A}$ is preferred to another alternative $\beta \in \mathcal{A}$, then the associated order relation $\rhd$ for preference model π satisfies $\alpha \rhd \beta$. This satisfaction relation can be extended to sets of preference statements in an obvious way. Let $\pi \in \mathcal{G}$ be a preference model and $\Gamma \subseteq \mathcal{L}$ a subset of preference statements, then we write $\pi \models \Gamma$, if π satisfies all preference statements in Γ, i.e., $\pi \models \varphi$ for all $\varphi \in \Gamma$.

Among the preference models $\mathcal{G}$, we can identify those that satisfy all of the user's preference statements in Γ. These are the preference models that are consistent with the user's preferences, and thus are the possible candidates for the unknown model by which the user compares alternatives. Let us denote this set of preference models by $\mathcal{G}_\Gamma$. Thus, $\mathcal{G}_\Gamma = \{\pi \in \mathcal{G} \mid \pi \models \Gamma\}$. In the problems considered in this dissertation, we aim to argue over the whole set of preference models $\mathcal{G}_\Gamma$. If all of the preference models (order relations) in $\mathcal{G}_\Gamma$ also satisfy another preference statement φ, then we know with certainty that the user agrees with the statement φ. In this case, we can deduce φ from Γ with certainty, written $\Gamma \models \varphi$.

Let us now formally define the Preference Deduction Problem.

Definition 3.1: Preference Deduction Problem

Let $\Gamma \subseteq \mathcal{L}$ and $\varphi \in \mathcal{L}$ be preference statements over a set of alternatives $\mathcal{A}$, and assume that the (unknown) user preference model is included in the set of preference models $\mathcal{G}$. Can we deduce the preference φ? More specifically, is it true that for all $\pi \in \mathcal{G}$ with $\pi \models \Gamma$, we have $\pi \models \varphi$?

The Preference Deduction problem thus depends on the set of alternatives $\mathcal{A}$, the set of preference models $\mathcal{G}$, and on the set of preference statements $\mathcal{L}$ that includes Γ and φ.

Alternatives can be specified by different variables (criteria) $\mathcal{V}$, so that an alternative is given as a vector of variable assignments. In most parts of this dissertation, we will assume that a set of variables $\mathcal{V}$ is given (see Chapters 5, 6, 7 and 8). However, in Chapter 4 we will allow alternatives to be abstract objects.

As mentioned before, preference models usually reflect the approach the user takes to compare or rate alternatives. In Section 2.5, we describe different assumptions that can be made on the user's preference model. In Chapters 5, 6, 7 and 8, we consider specific types of qualitative preference models. Here, the variable based view of alternatives allows us to assume preference models which do not give a preference value to alternatives, but instead compare alternatives on multiple criteria. Again, the exception is Chapter 4 in which only few assumptions are made on the user's preference model, which have to do with properties of the associated order relations rather than rules by which alternatives are compared or rated.

Finally, in this dissertation, we consider preference statements that are expressed by the user in a certain way, i.e., are included in a specified preference language $\mathcal{L}$. We discuss different common preference languages and semantics in Section 3.2 and define the comparative languages used in Chapters 5, 6, 7 and 8. Chapter 4 considers preference languages of (strongly) compositional statements. Here, (strong) compositionality is a property that is defined in connection with a set of preference models and can thus hold for many different types of preference statements.

Note that under the assumptions on preference language and preference model set, the Deduction Problem aims at making purely logic inferences. If the

assumptions are accurate, then the deduced preferences must hold. In contrast, preference learning approaches focus on learning one model in $\mathcal{G}$ that fits best with the given preferences and use it to make further deductions on the user's preferences [AD07, BCL$^+$10, BH12, CSS98, DIV07, FH10a, HFCB08, KZ10, LM09, SM06, YWLdJ10].

The Preference Deduction Problem does not aim to find an approximating model that best fits the user's preference statements, but argues over the whole set $\mathcal{G}_\Gamma$. It is thus crucial, to check if the set $\mathcal{G}_\Gamma$ is empty. That is, it is important to check if the given user statements are consistent, so that there exists a preference model that satisfies them. Otherwise, it would be possible to deduce any arbitrary preference statement. The Preference Consistency Problem decides if a set of given user preference statements is consistent, i.e., the statements do not contradict each other. It is strongly related to the Preference Deduction Problem, as under many assumptions of preference languages and models, the two problems are mutually expressive, see for example [BR07, MRW13, Wil09, Wil14] and our papers [GRW15, WGO17]. Here, by mutually expressiveness, we mean that an instance of the one problem can be solved by solving an instance of the other problem, and vice versa.

Formally the Preference Consistency Problem is defined as follows.

Definition 3.2: Preference Consistency Problem

Let $\Gamma \subseteq \mathcal{L}$ be a set of preference statements over a set of alternatives $\mathcal{A}$, and assume that the (unknown) user preference model is included in the set of preference models $\mathcal{G}$. Are the preference statements Γ consistent? More specifically, does there exist a preference model $\pi \in \mathcal{G}$ with $\pi \models \Gamma$?

The Preference Deduction and Preference Consistency Problem have been studied under different preference models, such as lexicographic models [Wil14], hierarchical models in our paper [WGO17], conditional lexicographic models [Wil09], Pareto orders in our paper [GW16], weighted sums in [BR07, MRW13, MW16, WRM15] and in our paper [GRW15], and on general strict total orders, as in e.g., work on conditional preference structures such as [BBD$^+$04a].

Previous work on preference inference based on standard lexicographic models have considered more restricted preference languages. Wilson [Wil14] consid-

ered only non-negated non-strict statements, which can only express that one assignment is at least as good as another (or equivalent). Kohli and Jedidi [KJ07] considers only non-negated strict statements, which can only express that one complete assignment is strictly better than another.

In our papers, efficient polynomial time algorithms to solve deduction and consistency have been developed for the cases of lexicographic models, a form of simple Pareto models and weighted sums, whereas considerations for hierarchical and more general Pareto models lead to NP-completeness and coNP-completeness results.

3.2 The Languages $\mathcal{L}^{\mathcal{A}}$, $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$

In some parts of this dissertation, we chose to focus on specific languages.

Sections 4.3.2.1 and 4.3.3.1, as well as Chapters 5, 6 and 7 are mainly concerned with strict and non-strict preference statements on complete assignments of variables (alternatives). We formally define a language over these statements in the following way.

Definition 3.3: The Language $\mathcal{L}^{\mathcal{A}}$

We define the set $\mathcal{L}^{\mathcal{A}}$ of strict and non-strict preference statements on outcomes $\mathcal{A}$ by $\mathcal{L}^{\mathcal{A}} = \{\alpha > \beta \mid \alpha, \beta \in \mathcal{A}\} \cup \{\alpha \geq \beta \mid \alpha, \beta \in \mathcal{A}\}$.

For some results, we need to consider the non-strict version $\varphi^{(\geq)}$ of a preference statement φ. Similarly, we can consider the non-strict version of a set of preferences Γ.

Definition 3.4: Non-Strict Versions of Statements $\mathcal{L}^{\mathcal{A}}$

If $\varphi \in \mathcal{L}^{\mathcal{A}}$ is a non-strict preference statement, then $\varphi^{(\geq)}$ is simply the statement φ. If $\varphi \in \mathcal{L}^{\mathcal{A}}$ is a strict preference statement $\alpha > \beta$, then we define $\varphi^{(\geq)}$ as $\alpha \geq \beta$. For a set of preferences $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, $\Gamma^{(\geq)}$ is defined to be the set $\{\varphi^{(\geq)} \mid \varphi \in \Gamma\}$.

We believe that statements of the language $\mathcal{L}^{\mathcal{A}}$ are simple to express for a user, as they do not necessarily require a deep analysis of the value assignments of

alternatives, but allow the users to use their intuition. However, one drawback of the language $\mathcal{L}^{\mathcal{A}}$ is that conditional dependencies between variables cannot be expressed compactly, instead a large set of preference statements is needed.

When working with user preferences, one searches for order relations on alternatives that satisfy the user's preferences statements. We define the following satisfaction relation for the previously described comparative preference statements. An order relation $\rhd$ on $\mathcal{A}$ is a binary relation on $\mathcal{A}$, and can thus be seen as a subset of tuples of $\mathcal{A} \times \mathcal{A}$. $\rhd$ is said to satisfy a non-strict statement $\alpha \geq \beta$ for outcomes α, β if $\alpha \rhd \beta$, i.e., $(\alpha, \beta) \in \rhd$. $\rhd$ satisfies strict statement $\alpha > \beta$ for outcomes α, β if $\alpha \rhd \beta$ and not $\beta \rhd \alpha$, i.e., $(\alpha, \beta) \in \rhd$ and $(\beta, \alpha) \notin \rhd$.

Mainly in Section 4.3 and Chapter 8, we analyse preference problems based on a relatively general language $\mathcal{L}_{pqT}$ of three types of hybrid preference statements $p \rhd q \mid T$ on partial assignments. These statements are a mix between ceteris paribus semantics and strong semantics (sometimes in connection with opportunistic semantics). Here, the ceteris paribus conditions are given by the set of variables T. If p and q are defined over the same variables $U \cup \{X\}$ and differ only on variable $X \in \mathcal{V}$ and $T = (\mathcal{V} \setminus U \setminus \{X\})$, then this form of statements can specify the structure of CP-nets [Wil09].

The *non-strict statement* $p \geq q \mid T$ expresses a non-strict relation under strong semantics, i.e., that all outcomes $\alpha \in \mathcal{A}$ that extend p are (non-strictly) preferred to outcomes $\beta \in \mathcal{A}$ that extend q given that α and β agree (i.e., have the same value assignment) on variables in T. Similarly, a *fully-strict statement* $p \gg q \mid T$ expresses a strict relation under strong semantics, i.e., that all outcomes $\alpha \in \mathcal{A}$ that extend p are strictly preferred to outcomes $\beta \in \mathcal{A}$ that extend q given that α and β agree on variables in T. *Weakly strict statements* $p > q \mid T$ express that $p \geq q \mid T$ under strong semantics and $p > q \mid T$ under something like an opportunistic semantics, i.e., all outcomes extending p are preferred to all outcomes extending q given they agree on T and there exists at least one outcome extending p that is strictly preferred to one outcome extending q and they agree on T. Note that both weakly strict and fully strict statements are the same usual strict order relation when p, q are complete assignments to the variables (and $T = \emptyset$).

Sections 4.3.2.1 and 4.3.3.1, as well as Chapter 8 are mainly concerned with fully strict, weakly strict and non-strict preference statements on partial assignments of variables. We formally define a language over these statements in the following way.

Definition 3.5: The Language $\mathcal{L}_{pqT}$

We define the set $\mathcal{L}_{pqT}$ of fully strict, weakly strict and non-strict preference statements on partial assignments by $\mathcal{L}_{pqT} = \{p \geq q \mid T,\ p > q \mid T,\ p \gg q \mid T$ such that $p \in \underline{P},\ q \in \underline{Q},\ P \cup Q \subseteq \mathcal{V},\ T \subseteq \mathcal{V} \setminus (P \cup Q)\}$.

While expressing preference statements of the language $\mathcal{L}_{pqT}$ most likely requires more cognitive effort (than statements in $\mathcal{L}^{\mathcal{A}}$), it enables the user to express more complicated structures like conditionality and also includes statements equivalent to $\mathcal{L}^{\mathcal{A}}$. This language consists of hybrids of different semantics: The fully strict statements correspond to a strong semantics on strict statements where a ceteris paribus condition is allowed. Weakly strict statements are a conjunction of strong semantics on non-strict preference statements and opportunistic semantics on strict statement — again a ceteris paribus condition can be expressed. Finally, non-strict statements in $\mathcal{L}_{pqT}$ correspond to a strong semantics on non-strict statements together with a ceteris paribus condition.

In order to simplify argumentations over the set of tuple of extending alternatives for p and q, we make the following definitions.

Definition 3.6: Tuples of Extending Alternatives for Statements in $\mathcal{L}_{pqT}$

For a preference statement $\varphi : p \rhd q \mid T$ in $\mathcal{L}_{pqT}$ with $\rhd \in \{\geq, >, \gg\}$, $p \in \underline{P}$ and $q \in \underline{Q}$, we define the set $\varphi^* = \{(\alpha, \beta) \in \mathcal{A}^2 \mid \alpha(P) = p,\ \beta(Q) = q,\ \alpha(T) = \beta(T)\}$. The set φ^*_A of all involved alternative tuples of φ that are fixed for $A \subseteq \mathcal{V}$ is then defined as $\varphi^*_A = \{(\alpha, \beta) \in \varphi^* \mid \alpha(A) = \beta(A)\}$.

For order relation $\rhd$ on $\mathcal{A}$ and preference statements on partial assignments we define the following. $\rhd$ satisfies the non-strict statement $p \geq q \mid T$, if $\varphi^* \subseteq \rhd$. The fully strict statement $p \gg q \mid T$ is satisfied by $\rhd$, if for all $(\alpha, \beta) \in \varphi^*$, $(\alpha, \beta) \in \rhd$ and $(\beta, \alpha) \notin \rhd$. Similarly, $\rhd$ satisfies the weakly strict statement $p > q \mid T$, if $\varphi^* \subseteq \rhd$ and there exists a tuple $(\alpha, \beta) \in \varphi^*$ such that $(\beta, \alpha) \notin \rhd$.

The preference language $\mathcal{L}'_{pqT}$ is an extension of $\mathcal{L}_{pqT}$ by certain negated statements. These are negations of non-strict preference statements $p \geq q \mid T$ on partial assignments, where $P = Q$. Note that only this negations is included in this language, since it has certain desirable properties that can be exploited.

Definition 3.7: The Language $\mathcal{L}'_{pqT}$

We define the language $\mathcal{L}'_{pqT}$ by $\mathcal{L}'_{pqT} = \mathcal{L}_{pqT} \cup \{\neg(p \geq q \mid T) \mid p \in \underline{P}, q \in \underline{Q}, P = Q, T \subseteq \mathcal{V} \setminus P\}$.

This language is quite expressive and possesses certain properties in connection with some tuples of compactly expressed order relations $\rhd$ that enable us to formulate fast and simple algorithms presented in this dissertation.

Languages (similar to) $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ have also been considered in, e.g., [BLW10, Wil09, Wil11]. Wilson [Wil04b, Wil04a] shows that such preferences are able to express CP-nets and TCP-nets. In [BLW10], instead of partial assignments logical formulas over a propositional language are compared and the condition can be given by a set of formulas.

Other relevant interpretations of statements on partial assignments p and q under condition T arise when one considers partial order relations on alternatives as opposed to total orders. The incomparability relation on alternatives $\alpha, \beta \in \mathcal{A}$ for the partial order $\rhd$, denoted by $\sim$, is given by $\alpha \sim \beta$ if and only if neither $\alpha \rhd \beta$ nor $\beta \rhd \alpha$ holds. Then the partial order $\rhd$ satisfies the fully incomparable statement, denoted $p \approx q \mid T$, if $\alpha \sim \beta$ for all $(\alpha, \beta) \in \varphi^*$. $\rhd$ satisfies the weakly incomparable statement, denoted $p \sim q \mid T$, if there exists $(\alpha, \beta) \in \varphi^*$ such that $\alpha \sim \beta$. The incomparability statement, denoted $p \times q \mid T$, is satisfied if there exists $(\alpha, \beta) \in \varphi^*$ such that $\alpha < \beta$ and $(\alpha', \beta') \in \varphi^*$ such that $\alpha' > \beta'$.

3.3 Pareto Models

In this Section, we extend our discussion of Pareto models from Section 2.5.2. Even though Pareto orders are widely studied in many fields, the idea of assuming Pareto orders as (unknown) user preference models for preference inference problems is novel. While we assume in our analysis of Pareto models in Chapter 5, that a fixed total value order is given for every variable, one could also consider more general Pareto models in which value orders for variables are not specified as in Section 4.3.3.

The simplest form of a Pareto model is a set $P = \{\{X_1\}, \ldots, \{X_k\}\}$ of singleton variable sets.

Definition 3.8: Fixed-Value-Order (FVO) Singleton Pareto Models

For variables $\mathcal{V}$ with fixed total value orders on variable domains, the set of fvo singleton Pareto models, which are sets of all-different singleton variable sets, is called $\mathcal{P}(1)$ and is in one-to-one correspondence to the power set of $\mathcal{V}$. The set of variables involved in a fvo singleton Pareto model $P = \{\{X_1\}, \ldots, \{X_k\}\}$ is denoted by $\sigma(P) = \{X_1, \ldots, X_k\} \subseteq \mathcal{V}$.

Note that since fvo singleton Pareto models include an arbitrary number of singleton variable sets, $\mathcal{P}(1)$ includes the empty model for which all alternatives are equivalent.

Formally, we can compare alternatives by these sets of variables in the following way.

Definition 3.9: FVO Singleton Pareto Order

Let P be a fvo Pareto model $P = \{\{X_1\}, \ldots, \{X_k\}\}$, and let $\geq_X$ be the fixed total value order for variable $X \in \mathcal{V}$. $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succcurlyeq_P \beta$, if and only if $\alpha(X) \geq_X \beta(X)$ for all variables X in P. $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$, written $\alpha \succ_P \beta$, if and only if $\alpha \succcurlyeq_P \beta$ and there exists a variable X in P such that $\alpha(X) >_X \beta(X)$.

As mentioned in Section 2.5.2, this order relation treads all variables involved as equally important and there is no compromise or tradeoff between different variables.

A generalisation of $\mathcal{P}(1)$ Pareto models can be constructed if tradeoffs within some sets of variables are allowed. Here, we assume that the variables have a common domain D, i.e., for all $X \in \mathcal{V}$, $\underline{X} = D$. We assume a commutative and associative operator $\oplus$ which acts on the variable's domain, such as the usual multiplication or addition on $\mathbb{R}$. With $\oplus$ we can combine arbitrarily many values of different variables. Note that this is a strong assumption, as this means that the variables and more general the features of the alternatives are commensurable and can be combined with $\oplus$.

Definition 3.10: k-bound Pareto Models

We define the set $\mathcal{P}(k)$ with $k \in \mathbb{N}$ to be the set of k-bound Pareto models that are sets of subsets of variables. More formally, $P \in \mathcal{P}(k)$ if $P = \{C_1, \ldots, C_l\}$ where $C_1, \ldots, C_l \subseteq \mathcal{V}$ with $|C_i| \leq k$ for all $i = 1, \ldots, l$ are non-empty, disjoint subsets of at most k variables. As before, the set of variables involved in a k-bound Pareto model $P = \{C_1, \ldots, C_l\}$ is denoted by $\sigma(P) = \bigcup_{i=1,\ldots,l} C_i$.

Note that for $k \leq c$ we have $\mathcal{P}(k) \subseteq \mathcal{P}(c)$. This property will be helpful later in the analysis for preference inference and consistency.

Definition 3.11: k-bound Pareto Order

Let $\oplus$ be a commutative and associative operator on the common variable's domain D and let $\geq$ be a fixed order relation on D. We compare alternatives by a k-bound Pareto model $P = \{C_1, \ldots, C_l\} \in \mathcal{P}(k)$ in the following way: $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succcurlyeq_P \beta$, if and only if $\bigoplus_{X \in C_i} \alpha(X) \geq \bigoplus_{X \in C_i} \beta(X)$ for all $i = 1, \ldots, l$. $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succ_P \beta$, if and only if $\alpha \succcurlyeq_P \beta$ and there exists $C_i \in P$ such that $\bigoplus_{X \in C_i} \alpha(X) > \bigoplus_{X \in C_i} \beta(X)$.

These models capture different situations. Imagine a choice between holiday alternatives. Here, different aspects of the alternatives can be put into categories, e.g., by costs (cost of hotel, flight, etc.), comfort (quality of hotel, transportation, etc.), time (length of stay, travel dates, season). Within these categories tradeoffs are allowed. Regarding the cost category, e.g., an expensive flight might be acceptable, if the hotel is cheap. This tradeoff is guaranteed by combining all values within one category by the operator $\oplus$. Note that to allow tradeoffs, variables have to be commensurable.

A holiday alternative is only then preferred to another, if it is preferred in every category, i.e., it is preferred in every one of the combined values of the variables within the categories. In a voting scenario, this would mean that voters are grouped together and aggregate their preferences expressed by combining their values with the operator $\oplus$. Then it is agreed that one candidate is preferred to another, if all groups agree on this unanimously.

Example 3.1

Consider the big family of Hal and Lois (the grandparents). They decide to buy and share a summer vacation house together with the families of their two sons Malcolm and Dewey. Let the preferences of the family be given by the following table, where 1 represents an acceptable alternative, and 0 a non-preferred alternative.

Name	House in Mountains	Old Farm House	Beach House
Hal	1	1	1
Lois	1	1	0
Malcolm	1	0	1
Malcolm's Wife	1	0	0
Dewey	0	1	1
Dewey's Wife	0	1	0
Malcolm's Son	0	0	1
Dewey's Daughter	0	0	0

One Pareto model could be given by {{Lois}, {Hal}, {Malcolm}, {Malcolm's wife}, {Dewey}, {Dewey's wife}}. This means all adults in the family get an equal vote, but the kids are left out of the choice of a holiday home. Under this model, the family cannot agree on any preference between the three houses, since for any pair of houses there are two adults with contradictory preferences.

Suppose, the three husbands make the decision on their own. The corresponding fvo Pareto model is given by P ={{Hal}, {Malcolm}, {Dewey}}. Even though Dewey and Malcolm disagree on the preference between the house in the mountains and the old farm house, the three men will agree unanimously that the beach house is better than any of the other houses. More formally, Beach House $\succcurlyeq_P$ House in Mountains and Beach House $\succcurlyeq_P$ Old Farm House.

Now consider the scenario where all parties (the grandparents, Malcolm's family and Dewey's family) agree to each take on a third of the costs, but in return expect an equal vote of their party in the choice of houses. The members of Malcolm's family have to aggregate their opinions on the different alternatives. Similarly, the other parties come to agreements on the rankings of the alternatives among each other. All parties vote with the goal to reach an unanimous decision. The corresponding k-bound Pareto model is given by {{Hal, Lois}, {Malcolm, Malcolm's wife, Malcolm's son}, {Dewey, Dewey's wife, Dewey's daughter}}. Let the operator $\oplus$, by which the preferences are aggregated, be the usual addition on the natural numbers, and consider the usual order relation on the natural numbers, i.e., the higher the number, the better. Then the aggregated preferences of the different parties are given in the following table.

Name	House in Mountains	Old Farm House	Beach House
The Grandparents	2	2	1
Malcolm's Family	2	0	2
Dewey's Family	0	2	1

Under this model, no decision can be made, since for every pair of houses there are two parties that disagree on their preferences.

Let us now define the aforementioned generalisation of fvo Pareto models to the case where the total value orders of the variable domains are not given but instead specified within a model. This implies a subjectivity of the user towards value orders of the variable domains. This type of models is relevant, where we are not considering variables with obvious value orders, but value orders that can differ from user to user. For any kind of purchase, only considering the price, users usually prefer cheaper options to more expensive ones. We can thus assume that the value order for costs respect the relation: the cheaper the better. However, for a visit to the cinema, it is not obvious what a user prefers for either of the different criteria: play time, genre and company. Is an afternoon show better than an evening one; does the user prefer comedy, drama or horror; does the user prefer to be accompanied by friends or a date? Here, it is reasonable to assume that the value orders vary from user to user, and should thus be specified in the user's preference model.

Definition 3.12: Changeable-Value-Order (CVO) Singleton Pareto Models

A cvo singleton Pareto model is a set of tuples $\{(X_1, \geq_{X_1}), \ldots, ((X_l, \geq_{X_l})\}$, where $\{X_1, \ldots, X_l\} \subseteq \mathcal{V}$ and the annotated order relations $\geq_{X_i}$ are total orders on the variable domains $\underline{X_i}$. We denote the set of cvo singleton Pareto models by $\mathscr{P}$.

The induced order relation on alternatives for a cvo singleton Pareto model $\mathscr{P}$ is similar to the order relation induced by fvo singleton Pareto models $\mathcal{P}(1)$.

Definition 3.13: CVO Singleton Pareto Order

Let $P = \{(X_1, \geq_{X_1}), \ldots, ((X_l, \geq_{X_l})\}$ be a cvo singleton Pareto model. Then P induces the following order relation on alternatives: $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succcurlyeq_P \beta$, if and only if $\alpha(X_i) \geq_i \beta(X_i)$ for all $(X_i, \geq_i) \in P$. $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under P, written $\alpha \succ_P \beta$, if and only if $\alpha \succcurlyeq_P \beta$ and there exists $(X_i, \geq_i) \in P$ such that $\alpha(X_i) >_i \beta(X_i)$.

Note that for convenience we use the same notation for the induced order of fvo and cvo singleton Pareto models. Again, we denote the variables involved in a cvo singleton Pareto model $P = \{(X_1, \geq_{X_1}), \ldots, ((X_l, \geq_{X_l})\}$ by $\sigma(P) = \{X_1, \ldots, X_l\}$.

3.4 Variable Hierarchies

We extend our considerations of lexicographic models from Section 2.5.2 by introducing three different types of models that rely on lexicographic orders.

Simple lexicographic models as defined in our paper [WGO15] depend on a subset of variables to be strictly ordered, i.e., no indifferences are allowed. Here, we assume that fixed value orders on the variable domains are given.

Definition 3.14: Fixed-Value-Order (FVO) Lexicographic Model

Formally we define a fvo lexicographic model L to be an ordered tuple $(X_1, \ldots, X_l)$ of a subset of all-different variables $\{X_1, \ldots, X_l\} \subseteq \mathcal{V}$. The set of fvo lexicographic models is denoted by $\mathcal{H}(1)$.

Assuming that for every variable $X \in \mathcal{V}$ there exists a fixed associated total value order $\geq_X$ on the domain of X, we can compare alternatives by such a tuple of variables in the following way:

Definition 3.15: FVO Lexicographic Order

Let $L = (X_1, \ldots, X_l)$ be a fvo lexicographic model, and let $\geq_X$ be the fixed total value order for variable $X \in \mathcal{V}$. Then L induces the following order relation on alternatives: $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under L, written $\alpha \succ_L \beta$, if and only if there exists $1 \leq i \leq l$ with $\alpha(X_i) >_{X_i} \beta(X_i)$ and for all $j < i$, $\alpha(X_j) = \beta(X_j)$. $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under L, written $\alpha \succcurlyeq_L \beta$, if and only if $\alpha \succ_L \beta$ or $\alpha(X_i) = \beta(X_i)$ for all $1 \leq i \leq l$.

We denote the variables involved in a fvo lexicographic model $L = (X_1, \ldots, X_l)$ by $\sigma(L) = \bigcup_{i=1,\ldots,l} X_i$. Given a fvo lexicographic model $L \in \mathcal{H}(1)$, we compare alternatives lexicographically by following the strict importance order on the variables $\sigma(L)$ involved.

We can define models that utilize a lexicographic order, however, only require a importance order on variables that is a partial order. Thus, these models are specified by an ordered tuple of sets of variables. The sets are called *levels* or *level sets* and contain variables of equal importance that can be combined by an operator. As in k-bound Pareto models, this allows tradeoffs between variables but at the price of enforcing variables to be commensurable and have the same domain D with a fixed value order $\geq$.

Definition 3.16: Hierarchical Models

A hierarchical model H is written as $H = (C_1, \ldots, C_l)$ where $C_1, \ldots, C_l \subseteq \mathcal{V}$ and the sets C_i are pairwise disjoint.

Similar as for fvo lexicographic models, we denote the variables involved in a

hierarchical model $H = (C_1, \ldots, C_l)$ by $\sigma(H) = \bigcup_{i=1,\ldots,l} C_i$. Given an associative and commutative operator $\oplus$ on the domain D of the variables $\mathcal{V}$ with value order $\geq$, a hierarchical model implies an order relation on the alternatives defined in the following way.

Definition 3.17: Hierarchical Order

Let $H = (C_1, \ldots, C_l)$ be a hierarchical preference model. Then $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under H, written $\alpha \succ_H \beta$, if and only if there exists $1 \leq i \leq l$ with $\bigoplus_{X \in C_i} \alpha(X) > \bigoplus_{X \in C_i} \beta(X)$ and for all $j < i$, $\bigoplus_{X \in C_j} \alpha(X) = \bigoplus_{X \in C_j} \beta(X)$. $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under H, written $\alpha \succcurlyeq_H \beta$, if and only if $\alpha \succ_H \beta$ or $\bigoplus_{X \in C_i} \alpha(X) = \bigoplus_{X \in C_i} \beta(X)$ for all $1 \leq i \leq l$.

As before for Pareto models, we can consider classes $\mathcal{H}(k)$ of hierarchical models with bounds on the sizes of level sets.

Definition 3.18: k-bound Hierarchical Models

Let $k \in \mathbb{N}$. $\mathcal{H}(k)$ is defined as the set $\{H = (C_1, \ldots, C_l) \mid H$ is a hierarchical model and $|C_i| \leq k \ \forall i = 1, \ldots, l\}$ which includes all hierarchical models with level sets of maximal cardinality k.

In this sense, we denote fvo lexicographic models by $\mathcal{H}(1)$ despite the slight abuse of notation, given that fvo lexicographic models are ordered tuples of variables as opposed to ordered tuples of singleton variable sets. Furthermore, for $k \leq c$ we have $\mathcal{H}(k) \subseteq \mathcal{H}(c)$. Note that in later chapters we explicitly demand monotonicity or strict monotonicity of the operator $\oplus$ in order to show some results.

Example 3.2

Consider the family decision of buying a holiday home as in the previous section. The Pareto models mentioned before all assume that every party/person has an equally important vote and thus all decisions have to be made unanimously. However, it might be that not every party pays an equal share or that some party spends much more time in the house. We could for example assume that the men decide amongst each other on the holi-

day houses as representatives of their families and that the grandparents pay 70% of the house. Thus the grandfather should have the most impact in this decision. Only if he is indifferent between two alternatives, Malcolm's preferences are considered, since he will spend most of his time in this house. Finally, in case both the grandfather and Malcolm are indifferent under two alternatives, Dewey's preferences can break the ties. This fvo lexicographic model is represented by $L =$ (Hal, Malcolm, Dewey). Considering the table of preferences given in the previous section, L implies Beach House $\succ_L$ House in Mountains $\succ_L$ Old Farm House.

Now consider the scenario where the preferences of all parties (the grandparents, Malcolm's family and Dewey's family) are aggregated by the operator $\oplus$, the usual addition on the natural numbers. Thus, all parties come to a decision together instead of having the men representing their preferences. For the reasons mentioned before, the grandparents have the most important say in the matter. Only if the grandparents are indifferent under some houses, Malcolm's family decides. And only if the grandparents and Malcolm's family are indifferent under two houses, then Dewey's family breaks the ties. This hierarchical model is given by $H =$ ({Hal, Lois}, {Malcolm, Malcolm's wife, Malcolm's son}, {Dewey, Dewey's wife, Dewey's daughter}) and the aggregated preferences are shown in the table below.

Level of Importance	Name	House in Mountains	Old Farm House	Beach House
1	The Grandparents	2	2	1
2	Malcolm's Family	2	0	2
3	Dewey's Family	0	2	1

Thus, H implies House in Mountains $\succ_H$ Old Farm House $\succ_H$ Beach House.

In addition to the just described variable hierarchies, we also consider a generalised form of lexicographic models. Here, a cvo lexicographic model again consists of a total order on a subset of variables. Additionally, variables in this tuple are annotated with a value order. This follows the assumption that value orders on variable domains are unknown and part of the preference model, implying the same subjectivity of value orders as in the models $\mathscr{P}$ as presented before.

Definition 3.19: Changeable-Value-Order (CVO) Lexicographic Models

A cvo lexicographic model π is a sequence of tuples $((X_1, \geq_{X_1}), \ldots, (X_l, \geq_{X_l}))$, where $\{X_1, \ldots, X_l\} \subseteq \mathcal{V}$ and the annotated order relations $\geq_{X_i}$ are total orders on the variable domains $\underline{X_i}$. We denote the set of cvo lexicographic models by $\mathscr{L}$.

We can now define an order relation on alternatives based on cvo lexicographic models similar to the order relation for fvo lexicographic models in the following way:

Definition 3.20: CVO Lexicographic Order

Let $\pi = ((X_1, \geq_{X_1}), \ldots, (X_l, \geq_{X_l}))$ be a cvo lexicographic model. Then π induces the following order relation on alternatives: $\alpha \in \mathcal{A}$ is strictly preferred to $\beta \in \mathcal{A}$ under π, written $\alpha \succ_\pi \beta$, if and only if there exists $1 \leq i \leq l$ with $\alpha(X_i) >_{X_i} \beta(X_i)$ and for all $j < i$ $\alpha(X_j) =_{X_j} \beta(X_j)$. $\alpha \in \mathcal{A}$ is preferred to $\beta \in \mathcal{A}$ under π, written $\alpha \succcurlyeq_\pi \beta$, if and only if $\alpha \succ_\pi \beta$ or $\alpha(X_i) =_{X_i} \beta(X_i)$ for all $1 \leq i \leq l$.

While the induced order relations for "simple" and cvo lexicographic models are very similar, they are not exactly the same as one uses predefined value orders for variable domains and the other deals with value orders that are specified by the model. Again, we denote the variables involved in a cvo lexicographic model $\pi = ((X_1, \geq_{X_1}), \ldots, ((X_l, \geq_{X_l}))$ by $\sigma(\pi) = \bigcup_{i=1,\ldots,l} X_i$. As for the fvo lexicographic models, a cvo lexicographic model π is an ordered tuple and thus represents a strict importance order on the variables $\sigma(\pi)$ involved.

Chapter 4

Strong Compositionality

In this chapter we introduce a general framework of inference. Here, we consider an arbitrary preference language $\mathcal{L}$ and an arbitrary finite set of preference models $\mathcal{G}$ that consists of order relations on the alternatives $\mathcal{A}$. That is, every $\pi \in \mathcal{G}$ is associated with an order relation $\succcurlyeq_\pi$ on $\mathcal{A}$. In this level of abstraction, it is not necessary to characterise alternatives by variable assignments as introduced in the previous chapter; they are simply abstract elements. We will introduce the property of (strong) compositionality for preference statements based on a set of preference models. This definition is based on the existence and properties of a composition operator that combines preference models. (Strong) compositionality then allows many statements about inference and consistency with very little restrictions on the setting. It is a property that many natural preference languages and models satisfy. Further statements about inference and consistency can then be made for strongly compositional languages considering specific preference model types.

This chapter is based on work in [WG17] where the concept of (strong) compositionality of preference statements has been first introduced in the context of CVO lexicographic models. In this chapter, we show that many of the results presented in [WG17] hold for the general case of arbitrary sets of preference models that are only restricted by very few assumptions. The generalised results rely on basic properties of the composition operator and new or modified definitions of minimal models, maximal models and minimal extensions.

In the first section, we introduce the definitions of composition operators and model extensions together with some basic properties. Here, we give examples of composition operators for lexicographic models and Pareto models to

which we will refer at a later point. In Section 4.2, we first introduce maximal models and the relaxed satisfaction relation $\models^*$, which expresses that a model can be extended to satisfy a preference statement. Based on this definition we are then able to define compositionality and strong compositionality for preference statement, Section 4.2.2. We describe more properties of (strong) compositionality and an algorithmic approach to solve the Consistency Problem for strongly compositional statements which could be efficient in many cases. In Section 4.3, we analyse the compositionality of statements in connection to specific types of preference models and give further properties, which lead to more details for the algorithmic approach. Here, we first consider preference models that can be associated with sets of variables in a certain way 4.3.1. We then specify the types of models even further to consider CVO lexicographic models (Section 4.3.2) and then CVO singleton Pareto models (Section 4.3.3). The last section concludes.

4.1 Composition and Extension

Compositional preference statements are defined via compositions and extensions of preference models. In this Section we present the necessary preliminaries to define (strong) compositionality of preference statements.

In order to consider compositions of preference models, we define a composition operator as an operator on a finite set (of preference models) $\mathcal{G}$ with three elementary properties.

Definition 4.1: Composition Operator

A composition operator is an operator $\circ : \mathcal{G} \times \mathcal{G} \longrightarrow \mathcal{G}$ on a finite set $\mathcal{G}$ that satisfies

1) $\pi \circ (\pi' \circ \pi'') = (\pi \circ \pi') \circ \pi''$ (associativity)

2) $\pi \circ \pi = \pi$ (idempotence)

3) If $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$, then $\pi_1 = \pi_2$. (asymmetry)

for all models $\pi_1, \pi_2, \pi, \pi', \pi'' \in \mathcal{G}$.

We define the associated extension relation of preference models for composi-

tion operators as follows.

Definition 4.2: Extension Relation

Let $\circ$ be a composition operator on a finite set $\mathcal{G}$. For $\pi, \pi' \in \mathcal{G}$ we say π' *extends* π if $\pi' \neq \pi$ and there exists a model $\pi'' \in \mathcal{G}$ such that $\pi' = \pi \circ \pi''$. We then write $\pi' \sqsupset \pi$, and write $\pi' \sqsupseteq \pi$ to mean that π' extends or equals π.

In the following, we show antisymmetry and transitivity properties for associated extension relations $\sqsupseteq$ of composition operators $\circ$. Using these properties we can then prove existence of "maximal models", which help us to formulate an approach to solve the Consistency Problem for strongly compositional preference statements later in this chapter. The next simple property follows easily from the definitions.

Lemma 4.1. *Let $\pi, \pi', \pi'' \in \mathcal{G}$ be preference models. If $\pi'' \sqsupseteq \pi'$, then $\pi \circ \pi'' \sqsupseteq \pi \circ \pi'$.*

Proof. If $\pi'' = \pi'$, then $\pi \circ \pi'' \sqsupseteq \pi \circ \pi'$ follows directly. If $\pi'' \sqsupset \pi'$, then there exists a model π_1 with $\pi'' = \pi' \circ \pi_1$. By applying associativity, we obtain $\pi \circ \pi'' = \pi \circ (\pi' \circ \pi_1) = (\pi \circ \pi') \circ \pi_1 \sqsupseteq \pi \circ \pi'$. □

The following results show that the extension relation $\sqsupseteq$ is antisymmetric and transitive on models $\mathcal{G}$.

Proposition 4.2. *The extension relation $\sqsupseteq$ is antisymmetric, i.e., for $\pi_1, \pi_2 \in \mathcal{G}$, if $\pi_1 \sqsupseteq \pi_2 \sqsupseteq \pi_1$ then $\pi_1 = \pi_2$.*

Proof. If $\pi_1 = \pi_2$, the statement trivially holds. Now suppose $\pi_1 \sqsupset \pi_2 \sqsupset \pi_1$. Then there exists $\pi, \pi' \in \mathcal{G}$ with $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. By asymmetry of composition operators, $\pi_1 = \pi_2$. □

Proposition 4.3. *The extension relation $\sqsupseteq$ is transitive, i.e., for $\pi_1, \pi_2, \pi_3 \in \mathcal{G}$, if $\pi_1 \sqsupseteq \pi_2 \sqsupseteq \pi_3$ then $\pi_1 \sqsupseteq \pi_3$.*

Proof. If $\pi_1 = \pi_2$ or $\pi_2 = \pi_3$, the $\pi_1 \sqsupseteq \pi_3$ obviously holds. Consider the case $\pi_1 \sqsupset \pi_2 \sqsupset \pi_3$. Then by definition of the extension relation, there exist models π_1' and π_2' in $\mathcal{G}$ with $\pi_1 = \pi_2 \circ \pi_1'$ and $\pi_2 = \pi_3 \circ \pi_2'$. Thus, using associativity of $\circ$, $\pi_1 = (\pi_3 \circ \pi_2') \circ \pi_1' = \pi_3 \circ (\pi_2' \circ \pi_1')$. Since $\pi_2' \circ \pi_1'$ is by definition of $\circ$ in $\mathcal{G}$, $\pi_1 \sqsupseteq \pi_3$. □

Proposition 4.4. *The extension relation $\sqsupset$ is transitive, i.e., for $\pi_1, \pi_2, \pi_3 \in \mathcal{G}$, if $\pi_1 \sqsupset \pi_2 \sqsupset \pi_3$ then $\pi_1 \sqsupset \pi_3$.*

Proof. Suppose that $\pi_1 \sqsupset \pi_2 \sqsupset \pi_3$. By transitivity of $\sqsupseteq$, we have $\pi_1 \sqsupseteq \pi_3$. It thus remains to show that $\pi_1 \neq \pi_3$. We can write $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_3 \circ \pi'$ for some models $\pi, \pi' \in \mathcal{G}$. Suppose $\pi_1 = \pi_3$. Then $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. By asymmetry of the composition operator, $\pi_1 = \pi_2$ which is a contradiction to $\pi_1 \sqsupset \pi_2$. Thus $\pi_1 \neq \pi_3$, i.e., $\pi_1 \sqsupset \pi_3$. □

Note that, associativity and idempotence of composition operators together with transitivity of $\sqsupset$ imply asymmetry. To prove this, suppose there would exist models $\pi_1, \pi_2, \pi, \pi' \in \mathcal{G}$ with $\pi_1 \neq \pi_2$ such that $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. Then $\pi_1 \sqsupset \pi_2 \sqsupset \pi_1$ and by transitivity $\pi_1 \sqsupset \pi_1$, which is a contradiction. Thus, given associativity and idempotence of a composition operator, transitivity of $\sqsupset$ and asymmetry are equivalent.

We can show that the composition with an extension is equal to the extension.

Lemma 4.5. *For $\pi, \pi' \in \mathcal{G}$, we have $\pi' \sqsupseteq \pi$ if and only if $\pi \circ \pi' = \pi'$.*

Proof. Suppose that $\pi = \pi'$. Then by idempotence of a composition, $\pi \circ \pi' = \pi' \circ \pi' = \pi'$. Now suppose π' extends π, i.e., $\pi' \sqsupset \pi$, and write π' as $\pi \circ \pi''$. Then, $\pi \circ \pi' = \pi \circ (\pi \circ \pi'')$ which by applying associativity and idempotence of compositions equals $\pi \circ \pi'' = \pi'$. Conversely, suppose that $\pi \circ \pi' = \pi'$. Then by definition, $\pi' \sqsupseteq \pi$. □

We now give examples of composition operators for two important types of preference models. These will be discussed in more detail at the end of the chapter.

Composition for Lexicographic Models Consider the set of fvo lexicographic models $\mathcal{H}(1)$ with fixed variable value orders as introduced in Definition 3.14 and 3.18. We can define a composition operator $\circ_{\mathcal{H}(1)}$ on $\mathcal{H}(1)$ in the following way. Let $\pi = (X_1, \ldots, X_k)$ and $\pi' = (X'_1, \ldots, X'_l)$ be two fvo lexicographic models. Let $\{X''_1, \ldots, X''_m\} = \sigma(\pi') \setminus \sigma(\pi)$ be the variables that appear in π' but not in π. Then the composition $\pi \circ_{\mathcal{H}(1)} \pi'$ is defined as the sequence $X_1, \ldots, X_k$ followed by all variables $\{X''_1, \ldots, X''_m\}$ in the same order as they appear in π'. If the model π for example is the sequence (airline, time) and the model π' is the

sequence (time, class, airline), then the composition $\pi \circ_{\mathcal{H}(1)} \pi'$ is the sequence (airline, time, class). Also, (airline, time, class) extends (airline, time).

We can show that this operator satisfies properties 1)–3) of compositions.

Proposition 4.6. *The operator $\circ_{\mathcal{H}(1)}$ as defined above for* fvo *lexicographic models $\mathcal{H}(1)$ is a composition operator.*

Proof. For simplicity, write $\circ_{\mathcal{H}(1)}$ as $\circ$. By definition, $\pi \circ \pi' \in \mathcal{H}(1)$ for models $\pi, \pi' \in \mathcal{H}(1)$. We now show that $\circ$ satisfies properties 1)–3) of compositions for fvo lexicographic models.
Associativity: Let $\pi_1 = (X_1, \ldots, X_k)$, $\pi_2 = (Y_1, \ldots, Y_l)$ and $\pi_3 = (Z_1, \ldots, Z_m)$ be three fvo lexicographic models in $\mathcal{H}(1)$. Let $(Y'_1, \ldots, Y'_{l'})$ be the sequence of variables in π_2 that do not appear in π_1, and let $(Z'_1, \ldots, Z'_{m'})$ be the sequence of variables in π_3 that do not appear in π_1 or π_2. Similarly, let $(Z''_1, \ldots, Z''_{m''})$ be the sequence of variables in π_3 that do not appear in π_2. Then $(\pi_1 \circ \pi_2) \circ \pi_3 = (X_1, \ldots, X_k, Y'_1, \ldots, Y'_{l'}, Z'_1, \ldots, Z'_{m'})$. Also, $\pi_1 \circ (\pi_2 \circ \pi_3) = (X_1, \ldots, X_k) \circ (Y_1, \ldots, Y_l, Z''_1, \ldots, Z''_{m''}) = (X_1, \ldots, X_k, Y'_1, \ldots, Y'_{l'}, Z'_1, \ldots, Z'_{m'})$.
Idempotence: Follows directly from the definition.
Asymmetry: Let $\pi_1 = (X_1, \ldots, X_k)$, $\pi_2 = (Y_1, \ldots, Y_l)$, and let π and π' be some fvo lexicographic models. Suppose, $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. Then π_1 begins with the sequence $(Y_1, \ldots, Y_l)$ of variables in π_2. Similarly π_2 begins with the sequence $(X_1, \ldots, X_k)$ of variables in π_1. Thus, $X_1 = Y_1$, $X_2 = Y_2$, etc. and $l = k$. Hence, $\pi_1 = \pi_2$. □

We can define a composition operator for cvo lexicographic models $\mathscr{L}$ (see Definition 3.19) where variable value orders are not fixed (but part of the model) in the following way. Let $\pi = ((X_1, \geq_1), \ldots, (X_k, \geq_k))$ and $\pi' = ((X'_1, \geq'_1), \ldots, (X'_l, \geq'_l))$ be two cvo lexicographic models in $\mathscr{L}$. Let $\{X''_1, \ldots, X''_m\} = \sigma(\pi') \setminus \sigma(\pi)$ be the variables that appear in π' but not in π. Then the composition $\pi \circ_{\mathscr{L}} \pi'$ is defined as the sequence of tuples in π followed by all tuples with variables $\{X''_1, \ldots, X''_m\}$ in the same order as they appear in π', i.e., $\pi \circ_{\mathscr{L}} \pi' = ((X_1, \geq_1), \ldots, (X_k, \geq_k), (X''_1, \geq''_1), \ldots, (X''_m, \geq''_m))$. If the model π for example is the sequence ((airline, KLM > LAN), (time, day > night)) and the model π' is the sequence ((time, night > day), (class, economy > business), (airline, LAN > KLM)), then the composition $\pi \circ \pi'$ is the sequence ((airline, KLM > LAN), (time, day > night), (class, economy > business)). The proof showing that this is a composition operator for cvo lexicographic models is similar to the proof of Proposition 4.6.

Lemma 4.7. *The operator $\circ_{\mathscr{L}}$ for* cvo *lexicographic models $\mathscr{L}$ is a composition operator.*

Proof. For simplicity, write $\circ_{\mathscr{L}}$ as $\circ$. By definition, $\pi \circ \pi' \in \mathscr{L}$ for $\pi, \pi' \in \mathscr{L}$. We now show that $\circ$ satisfies properties 1)–3) of compositions for cvo lexicographic models.
Associativity: Let $\pi_1 = ((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}))$, $\pi_2 = ((Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l}))$ and $\pi_3 = ((Z_1, \geq_{Z_1}), \ldots, (Z_m, \geq_{Z_m}))$ be three cvo lexicographic models. Furthermore, let $(Y'_1, \ldots, Y'_{l'})$ be the sequence of variables in π_2 that do not appear in π_1 and let $(Z'_1, \ldots, Z'_{m'})$ be the sequence of variables in π_3 that do not appear in π_1 or π_2. Similarly, let $(Z''_1, \ldots, Z''_{m''})$ be the sequence of variables in π_3 that do not appear in π_2. Then $(\pi_1 \circ \pi_2) \circ \pi_3 = ((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}), (Y'_1, \geq_{Y'_1}), \ldots, (Y'_{l'}, \geq_{Y'_{l'}}), (Z'_1, \geq_{Z'_1}), \ldots, (Z'_{m'}, \geq_{Z'_{m'}}))$. Also, we have that $\pi_1 \circ (\pi_2 \circ \pi_3) = ((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k})) \circ ((Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l}), (Z''_1, \geq_{Z''_1}), \ldots, (Z''_{m''}, \geq_{Z''_{m''}}))$ $= ((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}), (Y'_1, \geq_{Y'_1}), \ldots, (Y'_{l'}, \geq_{Y'_{l'}}), (Z'_1, \geq_{Z'_1}), \ldots, (Z'_{m'}, \geq_{Z'_{m'}}))$.
Idempotence: Follows directly from the definition.
Asymmetry: Let $\pi_1 = ((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}))$, $\pi_2 = ((Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l}))$, π and π' be some cvo lexicographic models in $\mathscr{L}$. Suppose, $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. Then by definition of composition operators, π_1 begins with the sequence $((Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l}))$ of tuples in π_2. Similarly, π_2 begins with the sequence $((X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}))$ of tuples in π_1. Thus, $(X_1, \geq_{X_1}) = (Y_1, \geq_{Y_1})$, $(X_2, \geq_{X_2}) = (Y_2, \geq_{Y_2})$, etc. and $l = k$. Hence, $\pi_1 = \pi_2$. □

Composition for Singleton Pareto Models We consider the set of fvo Pareto models $\mathcal{P}(1)$ as introduced in Section 3.3. A composition operator $\circ_{\mathcal{P}(1)}$ on $\mathcal{P}(1)$ can be defined in a similar way to the composition $\circ_{\mathcal{H}(1)}$ for fvo lexicographic models. Let $\pi = \{X_1, \ldots, X_k\}$ and $\pi' = \{X'_1, \ldots, X'_l\}$ be two fvo Pareto models. Then the composition $\pi \circ_{\mathcal{P}(1)} \pi'$ is defined as the union of π and π', $\{X_1, \ldots, X_k\} \cup \{X'_1, \ldots, X'_l\}$.

We can show that this composition operator satisfies properties 1)–3).

Proposition 4.8. *The operator $\circ_{\mathcal{P}(1)}$ for* fvo *singleton Pareto models is a composition operator.*

Proof. For simplicity, write $\circ_{\mathcal{P}(1)}$ as $\circ$. By definition, $\pi \circ \pi' \in \mathcal{P}(1)$ for $\pi, \pi' \in \mathcal{P}(1)$. We now show that $\circ$ satisfies the properties of composition operators for fvo singleton Pareto models.

Associativity: Let $\pi_1 = \{X_1, \ldots, X_k\}$, $\pi_2 = \{Y_1, \ldots, Y_l\}$ and $\pi_3 = \{Z_1, \ldots, Z_m\}$ be three fvo singleton Pareto models. Then because of associativity of the set union, $(\pi_1 \circ \pi_2) \circ \pi_3 = (\{X_1, \ldots, X_k\} \cup \{Y_1, \ldots, Y_l\}) \cup \{Z_1, \ldots, Z_m\} = \{X_1, \ldots, X_k\} \cup (\{Y_1, \ldots, Y_l\} \cup \{Z_1, \ldots, Z_m\}) = \pi_1 \circ (\pi_2 \circ \pi_3)$.
Idempotence: Follows directly from the definition.
Asymmetry: Let $\pi_1 = \{X_1, \ldots, X_k\}$, $\pi_2 = \{Y_1, \ldots, Y_l\}$, π and π'' be some fvo singleton Pareto models. Suppose, $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. Then $\pi_1 = \pi_2 \cup \pi$ and in particular $\pi_2 \subseteq \pi_1$. Similarly $\pi_2 = \pi_1 \cup \pi'$ and thus $\pi_1 \subseteq \pi_2$. Hence, $\pi_1 = \pi_2$. □

As for the fvo lexicographic case, the definition of a composition operator for fvo singleton Pareto models can be extended to more general cvo singleton models, where value orders of the variables are part of the models. Here, if the variable X appears in the tuple $(X, \geq)$ in π and in the tuple $(X, \geq')$ in π', then we define the composition $\pi \circ_{\mathscr{P}} \pi'$ to contain only the tuple $(X, \geq)$, not $(X, \geq')$. More detailed, for cvo singleton Pareto models $\pi = \{(X_1, \geq_1), \ldots, (X_k, \geq_k)\}$ and $\pi' = \{(X'_1, \geq'_1), \ldots, (X'_l, \geq'_l)\}$ with $\{X''_1, \ldots, X''_m\} = \sigma(\pi') \setminus \sigma(\pi)$, the composition $\pi \circ_{\mathscr{P}} \pi'$ is defined as $\pi \circ_{\mathscr{P}} \pi' = \{(X_1, \geq_1), \ldots, (X_k, \geq_k), (X''_1, \geq''_1), \ldots, (X''_m, \geq''_m)\}$.

The proof showing that this is a composition operator for cvo singleton Pareto models $\mathscr{P}$ is similar to the proof of Lemma 4.7 and Proposition 4.8.

Lemma 4.9. *The operator $\circ_{\mathscr{P}}$ for* cvo *singleton Pareto models $\mathscr{P}$ is a composition operator.*

Proof. For simplicity write the operator $\circ_{\mathscr{P}}$ as $\circ$.
Associativity: Let $\pi_1 = \{(X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k})\}$, $\pi_2 = \{(Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l})\}$, and $\pi_3 = \{(Z_1, \geq_{Z_1}), \ldots, (Z_m, \geq_{Z_m})\}$ be three cvo singleton Pareto models in $\mathscr{P}$. Let $\{Y'_1, \ldots, Y'_{l'}\}$ be the set of variables in π_2 that do not appear in π_1 and let $\{Z'_1, \ldots, Z'_{m'}\}$ be the set of variables in π_3 that do not appear in π_1 or π_2. Similarly, let $\{Z''_1, \ldots, Z''_{m''}\}$ be the set of variables in π_3 that do not appear in π_2. Then $(\pi_1 \circ \pi_2) \circ \pi_3 = \{(X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}), (Y'_1, \geq_{Y'_1}), \ldots, (Y'_{l'}, \geq_{Y'_{l'}}), (Z'_1, \geq_{Z'_1}), \ldots, (Z'_{m'}, \geq_{Z'_{m'}})\}$.
Also, $\pi_1 \circ (\pi_2 \circ \pi_3) = \{(X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k})) \circ ((Y_1, \geq_{Y_1}), \ldots, (Y_l, \geq_{Y_l}), (Z''_1, \geq_{Z''_1}), \ldots, (Z''_{m''}, \geq_{Z''_{m''}})\} = \{(X_1, \geq_{X_1}), \ldots, (X_k, \geq_{X_k}), (Y'_1, \geq_{Y'_1}), \ldots, (Y'_{l'}, \geq_{Y'_{l'}}), (Z'_1, \geq_{Z'_1}), \ldots, (Z'_{m'}, \geq_{Z'_{m'}})\}$.
Idempotence: Follows directly from the definition.
Asymmetry: Let π_1, π_2, π and π'' be some cvo singleton Pareto models. Suppose, $\pi_1 = \pi_2 \circ \pi$ and $\pi_2 = \pi_1 \circ \pi'$. Then $\pi_2 \subseteq \pi_1$ and $\pi_1 \subseteq \pi_2$. Hence, $\pi_1 = \pi_2$. □

4.2 Inference and Strong Compositionality

In this section, we use the definitions of a composition operator $\circ$ and an associated extension relation $\sqsupseteq$ to define a relaxed version of inference and the notion of maximal and minimal models. We are then finally able to define *strong compositionality*, a property of preference statements in connection with a set of preference models. More specifically, a preference statement φ is strongly compositional if the composition of two models satisfies φ whenever the second and some extension of the first satisfy φ. Strong compositionality induces many properties for preference inference which are discussed in detail later in this chapter and can lead to efficient computations.

Throughout this section, we consider some language $\mathcal{L}$, and satisfaction relation $\models \subseteq \mathcal{G} \times \mathcal{L}$. Thus, $\mathcal{L}$ is a set of preference statements and the relation $\models$ determines when a preference model in $\mathcal{G}$ satisfies a statement as introduced in Section 8.3.

4.2.1 The Induced Relation $\models^*$ and Maximal Models

Recall from Section 3.1 that the inference relation $\Gamma \models_{\mathcal{G}} \varphi$ holds if for all $\pi \in \mathcal{G}$ such that $\pi \models \Gamma$ we have $\pi \models \varphi$.

From the relation $\models$ together with a composition operator we define the derived relation $\models^*$ as follows.

Definition 4.3: $\models^*$-Satisfaction Relation

Let $\pi \in \mathcal{G}$ and $\varphi \in \mathcal{L}$. Then $\pi \models^* \varphi$ if and only if there exists $\pi' \in \mathcal{G}$ either extending or equalling π, i.e., $\pi' \sqsupseteq \pi$, such that $\pi' \models \varphi$.

Thus, $\pi \models^* \varphi$ holds either if π satisfies φ or some extension of π satisfies φ, i.e., there exists a model $\pi'' \in \mathcal{G}$ such that $\pi \circ \pi'' \models \varphi$. We extend the relation $\models^*$ to sets of statements in the usual way: for $\Gamma \subseteq \mathcal{L}$, define $\pi \models^* \Gamma$ if and only if $\pi \models^* \varphi$ holds for every $\varphi \in \Gamma$. The following lemma follows easily from the definitions and shows essential properties of $\models^*$ that will be useful later on.

Lemma 4.10. *Let $\pi, \pi' \in \mathcal{G}$ and $\Gamma \subseteq \mathcal{L}$.*

(i) $\pi \models \Gamma \Rightarrow \pi \models^* \Gamma$.

(ii) Suppose that π' extends π. Then $\pi' \models^* \Gamma \Rightarrow \pi \models^* \Gamma$.

Proof. (i) follows immediately from the definition of $\models^*$. Regarding (ii), $\pi' \models^* \varphi$ for $\varphi \in \Gamma$ implies that there exists π'' with $\pi'' \sqsupseteq \pi'$ and $\pi'' \models \varphi$. By the transitivity of $\sqsupseteq$ (Proposition 4.3), $\pi'' \sqsupseteq \pi$, and thus, $\pi \models^* \varphi$ for every $\varphi \in \Gamma$. □

We now define maximal models for $\models$ and $\models^*$, which play a crucial role in our algorithmic solution for the Preference Consistency problem with strongly compositional statements. They are furthermore important for many inference results.

Definition 4.4: Maximal Models

We say that $\pi \in \mathcal{G}$ is a *maximal model of* Γ if $\pi \models \Gamma$ and for all extending models $\pi' \sqsupset \pi$ we have $\pi' \not\models \Gamma$.

In particular, if there exists no strict extension $\pi' \sqsupset \pi$ for $\pi \in \mathcal{G}$, then π is a maximal model of Γ if and only if $\pi \vDash \Gamma$. Note that the definition of maximal models depends on an extension relation and thus on a composition operator.

Example 4.1

Consider the set of fvo lexicographic models $\mathcal{H}(1)$ that compare flight connections by variables $\mathcal{V} = \{$airline, class, time$\}$ and fixed value orders KLM > LAN, business > economy and day > night. Then $\pi =$ (airline, time, class) satisfies $\Gamma =$ { (KLM, economy, night) $>$ (LAN, business, day) } because airline is the first variable in the sequence of π and KLM > LAN. Also, π is a maximal model of Γ because there are no variables in $\mathcal{V}$ left to extend the model. For $\Gamma' =$ { (KLM, economy, day) $>$ (LAN, business, night), (LAN, business, night) $\geq$ (LAN, business, day)}, the model (airline, class) is a maximal model as the remaining variable, time, cannot be added to the model without violating the second preference statement in Γ' since day > night.

Under our general assumption that $\mathcal{G}$ is a finite set of preference models, we can show that there always exists a maximal model for consistent $\Gamma \subseteq \mathcal{L}$.

Proposition 4.11. *Let $\Gamma \subseteq \mathcal{L}$ be a set of consistent preference statements and $\mathcal{G}$ a finite set of preference models. Then for $\pi \in \mathcal{G}$ such that $\pi \models \Gamma$, either π is a maximal model of Γ or there exists a maximal model of Γ that extends π. In particular, there exists a model $\pi \in \mathcal{G}$ that is a maximal model of Γ.*

Proof. Let $\pi_1 \in \mathcal{G}$ be a model of Γ. Assume π_1 is not a maximal model of Γ. Then, there exists a model $\pi_2 \in \mathcal{G}$ with $\pi_2 \sqsupset \pi_1$ and $\pi_2 \models \Gamma$. If π_2 is not a maximal model of Γ, the previous argument can be applied again. Thus there exists a sequence $\pi_1 \sqsubset \pi_2 \sqsubset \pi_3 \sqsubset \dots$ of models in $\mathcal{G}$ that satisfy Γ. Since the set of models $\mathcal{G}$ is assumed to be finite, every such sequence of extensions is either finite and cannot be extended further, or there exist models π_i, π_j with $i < j$ in the sequence such that $\pi_i = \pi_j$. By Proposition 4.4, $\sqsubset$ is transitive, and thus $\pi_k \sqsupset \pi_l$ for all $l < k$. Thus, there cannot exist π_i, π_j with $i < j$ and $\pi_i = \pi_j$. Hence, there exists a model $\pi_k \in \mathcal{G}$ of Γ such that $\pi_1 \sqsubset \pi_2 \sqsubset \pi_3 \sqsubset \dots \sqsubset \pi_k$ and there exists no extension of π_k that is a model of Γ. Then π_k is a maximal model, and $\pi_1 \sqsubset \pi_k$ (again by transitivity of $\sqsupset$).

Since Γ is consistent, there exists a model π of Γ and hence, there exists a maximal model (extending or equalling π) of Γ. □

Analogously to maximal models of Γ, we define maximal $\models^*$-models of Γ.

Definition 4.5: Maximal $\models^*$-Models

A maximal $\models^*$-model of Γ is an element $\pi \in \mathcal{G}$ such that $\pi \models^* \Gamma$ and for all extending models $\pi' \sqsupset \pi$ we have $\pi' \not\models^* \Gamma$.

For any consistent Γ and finite model set $\mathcal{G}$, there always exists a maximal $\models^*$-model. The proof is similar to the proof of existence of a maximal model of Γ, see Proposition 4.11.

Proposition 4.12. *Let $\Gamma \subseteq \mathcal{L}$ be a set of consistent preference statements and assume that $\mathcal{G}$ is a finite set of preference models. Then for $\pi \in \mathcal{G}$ such that $\pi \models^* \Gamma$, either π is a maximal $\models^*$-model of Γ or there exists a maximal $\models^*$-model of Γ that extends π. In particular, there exists a model $\pi \in \mathcal{G}$ that is a maximal $\models^*$-model of Γ. Furthermore, for any $modelsstar$-model of Γ, there exists a model $\pi' \in \mathcal{G}$ with $\pi' \sqsupseteq \pi$ that is a maximal model of Γ.*

Proof. Let $\pi_1 \in \mathcal{G}$ with $\pi_1 \models^* \Gamma$. Assume π_1 is not a maximal $\models^*$-model of Γ.

Then, there exists a model $\pi_2 \in \mathcal{G}$ with $\pi_2 \sqsupset \pi_1$ and $\pi_2 \models^* \Gamma$. If π_2 is not a maximal $\models^*$-model of Γ, the previous argument can be applied again.

Thus there exists a sequence $\pi_1 \sqsubset \pi_2 \sqsubset \pi_3 \sqsubset \dots$ of models in $\mathcal{G}$ that $\models^*$-satisfy Γ.

Since the set of models $\mathcal{G}$ is assumed to be finite, every such sequence of extensions is either finite and cannot be extended further, or there exist models π_i, π_j with $i < j$ in the sequence such that $\pi_i = \pi_j$. By Proposition 4.4, $\sqsubset$ is transitive, and thus $\pi_k \sqsupset \pi_l$ for all $l < k$. Thus, there cannot exist π_i, π_j with $i < j$ and $\pi_i = \pi_j$. Hence, there exists a $\models^*$-model $\pi_k \in \mathcal{G}$ of Γ such that $\pi_1 \sqsubset \pi_2 \sqsubset \pi_3 \sqsubset \dots \sqsubset \pi_k$ and there exists no extension of π_k that is a $\models^*$-model of Γ. Then π_k is a maximal $\models^*$-model of Γ, and $\pi_1 \sqsubset \pi_k$ (again by transitivity of $\sqsupset$).

If π_k is a maximal $\models^*$-model of Γ extending π_1, then (by definition of $\models^*$) there exists a model $\pi \models \Gamma$ with $\pi \sqsupseteq \pi_k$. By Proposition 4.11, there exists a maximal model π' of Γ that extends π. Thus $\pi' \sqsupseteq \pi \sqsupseteq \pi_k \sqsupseteq \pi_1$ and by transitivity of $\sqsupseteq$, we have $\pi' \sqsupseteq \pi_1$. □

4.2.2 (Strongly) Compositional Preference Statements

We are now finally able to define (strong) compositionality, a property of preference statements that has strong implications regarding inference and consistency.

Definition 4.6: (Strong) Compositionality

Let $\varphi \in \mathcal{L}$. We say that φ is *compositional* if for all $\pi, \pi' \in \mathcal{G}$, $\pi \models \varphi$ and $\pi' \models \varphi$ implies $\pi \circ \pi' \models \varphi$.

We say that φ is *strongly compositional* if for all $\pi, \pi' \in \mathcal{G}$, $\pi \models^* \varphi$ and $\pi' \models \varphi$ implies $\pi \circ \pi' \models \varphi$.

For $\Gamma \subseteq \mathcal{L}$, we define Γ to be *compositional* if every element of Γ is compositional. Similarly, we say that Γ is *strongly compositional* if every element of Γ is strongly compositional.

Note that if $\varphi \in \mathcal{L}$ is inconsistent, i.e., there exists no $\pi \in \mathcal{G}$ with $\pi \models \varphi$, then φ is trivially (strongly) compositional.

Example 4.2

Consider the set of fvo lexicographic models $\mathcal{H}(1)$ in the flight connection example with variables and fixed value orders as before (see Example 4.1). The preference statements $\Gamma = \{$(KLM, economy, day) $>$ (LAN, business, night), (LAN, business, night) $\geq$ (LAN, business, day)$\}$ are strongly compositional. A more general proof of the compositionality of strict and non-strict preference statements for cvo lexicographic models will be given later in Section 4.3.2.

The definitions immediately imply the following relations for sets of preference statements which we will use often in later proofs.

Lemma 4.13. *Let $\Gamma \subseteq \mathcal{L}$, and let $\pi, \pi' \in \mathcal{G}$.*

- *If Γ is strongly compositional then it is compositional.*
- *If Γ is compositional then $\pi \models \Gamma$ and $\pi' \models \Gamma$ imply $\pi \circ \pi' \models \Gamma$.*
- *If Γ is strongly compositional then $\pi \models^* \Gamma$ and $\pi' \models \Gamma$ imply $\pi \circ \pi' \models \Gamma$.*

Proof.

- Assume Γ is strongly compositional. Let $\pi \models \varphi$ and $\pi' \models \varphi$ for some $\varphi \in \Gamma$. Then also $\pi \models^* \varphi$, by Lemma 4.10. By strong compositionality of Γ, $\pi \circ \pi' \models \varphi$. Thus, every $\varphi \in \Gamma$ is compositional, i.e., Γ is compositional.
- Assume Γ is compositional and $\pi \models \Gamma$ and $\pi' \models \Gamma$. Then for every $\varphi \in \Gamma$, $\pi \models \varphi$ and $\pi' \models \varphi$ and thus by compositionality $\pi \circ \pi' \models \varphi$. Hence, $\pi \circ \pi' \models \Gamma$.
- Assume Γ is strongly compositional and $\pi \models^* \Gamma$ and $\pi' \models \Gamma$. Then for every $\varphi \in \Gamma$, $\pi \models^* \varphi$ and $\pi' \models \varphi$ and thus by strong compositionality $\pi \circ \pi' \models \varphi$. Hence, $\pi \circ \pi' \models \Gamma$.

□

Although being strongly compositional might appear to be quite a restrictive assumption, it turns out that it is satisfied by many natural preference statements, as illustrated in Section 4.3. In the following, we give a lemma which, roughly speaking, states that the property of being [strongly] compositional is closed under conjunction.

Lemma 4.14. *Let $\Gamma \subseteq \mathcal{L}$ and let $\psi \in \mathcal{L}$. Suppose that ψ is such that for all $\pi \in \mathcal{G}$, $\pi \models \psi \iff \pi \models \Gamma$. If Γ is compositional then ψ is compositional. If Γ is strongly compositional then ψ is strongly compositional and, for all $\pi \in \mathcal{G}$, [$\pi \models^* \psi \iff \pi \models^* \Gamma$ and Γ is consistent].*

Proof. Suppose that Γ is compositional. Consider any $\pi, \pi' \in \mathcal{G}$ with $\pi, \pi' \models \psi$. Then, $\pi, \pi' \models \Gamma$, which, since Γ is compositional, implies $\pi \circ \pi' \models \Gamma$, and thus, $\pi \circ \pi' \models \psi$, showing that ψ is compositional.

Now assume for the remainder of the proof that Γ is strongly compositional. First suppose also that $\pi \models^* \psi$. Then there exists π'' such that $\pi'' \sqsupseteq \pi$ and $\pi'' \models \psi$, which implies that ψ is consistent. Also, $\pi'' \models \Gamma$, and so Γ is consistent. For all $\varphi \in \Gamma$ we have $\pi'' \models \varphi$ and thus $\pi \models^* \varphi$. We have shown that $\pi \models^* \Gamma$.

For the converse, we assume that $\pi \models^* \Gamma$ and Γ is consistent. Consistency of Γ implies that there exists $\pi' \in \mathcal{G}$ such that $\pi' \models \Gamma$. Since Γ is strongly compositional, $\pi \circ \pi' \models \Gamma$, and so, $\pi \circ \pi' \models \psi$ which implies that $\pi \models^* \psi$, because $\pi \circ \pi' \sqsupseteq \pi$.

Finally, we show that ψ is strongly compositional. Consider any $\pi, \pi' \in \mathcal{G}$ with $\pi \models^* \psi$ and $\pi' \models \psi$. We have, by the earlier part, that $\pi \models^* \Gamma$, and also $\pi' \models \Gamma$. Strong compositionality of Γ implies $\pi \circ \pi' \models \Gamma$ and thus, $\pi \circ \pi' \models \psi$, proving that ψ is strongly compositional. □

When a preference statement $\varphi \in \mathcal{L}$ is strongly compositional then there is a further simple composition property just involving $\models^*$, as expressed by the following lemma.

Lemma 4.15. *Let $\varphi \in \mathcal{L}$, and let $\pi, \pi' \in \mathcal{G}$. If φ is strongly compositional then $\pi \models^* \varphi$ and $\pi' \models^* \varphi$ implies $\pi \circ \pi' \models^* \varphi$. Also, if $\Gamma \subseteq \mathcal{L}$ is strongly compositional then $\pi \models^* \Gamma$ and $\pi' \models^* \Gamma$ implies $\pi \circ \pi' \models^* \Gamma$.*

Proof. Assume that $\pi \models^* \varphi$ and $\pi' \models^* \varphi$, so there exists π'' with $\pi'' \sqsupseteq \pi'$ and $\pi'' \models \varphi$. If φ is strongly compositional then $\pi \circ \pi'' \models \varphi$, and thus, $\pi \circ \pi' \models^* \varphi$, since by Lemma 4.1 $\pi \circ \pi'' \sqsupseteq \pi \circ \pi'$. The second part follows immediately from the first. □

4.2.3 Consistency of Strongly Compositional Preferences

We will now consider the consistency of sets of strongly compositional preference statements and formulate a key theorem that forms the basis for our algorithmic approach to solve the Consistency Problem.

The last point of Lemma 4.13 implies that, for strongly compositional and consistent Γ, if $\pi \models^* \Gamma$ then there exists a model of Γ either equalling or extending π. In fact we have:

Lemma 4.16. *Suppose that Γ is strongly compositional, and let π be an element of $\mathcal{G}$. Then [there exists $\pi' \in \mathcal{G}$ with $\pi' \sqsupseteq \pi$ and $\pi' \models \Gamma$] if and only if [Γ is consistent and $\pi \models^* \Gamma$].*

Proof. $\Rightarrow$: First assume that there exists $\pi' \in \mathcal{G}$ with $\pi' \sqsupseteq \pi$ and $\pi' \models \Gamma$. Clearly, Γ is consistent. Consider any $\varphi \in \Gamma$. We have $\pi' \models \varphi$, which implies $\pi \models^* \varphi$. Therefore, $\pi \models^* \Gamma$.

$\Leftarrow$: Assume that Γ is consistent and $\pi \models^* \Gamma$. Then there exists $\pi' \in \mathcal{G}$ with $\pi' \models \Gamma$. Since Γ is strongly compositional, $\pi \circ \pi' \models \Gamma$, by Lemma 4.13, and we have $\pi \circ \pi' \sqsupseteq \pi$. □

The following result states that if Γ is strongly compositional the maximal $\models^*$-models satisfy exactly the same elements of Γ. Also, for consistent Γ, all maximal models of Γ are $\models^*$-maximal models of Γ and vice versa.

Theorem 4.1: Maximal Model Satisfaction of Preferences

Assume that $\Gamma \subseteq \mathcal{L}$ is strongly compositional. Then, the following hold.

(i) For maximal $\models^*$-models π, π' of Γ and for $\varphi \in \Gamma$, $\pi \models \varphi \iff \pi' \models \varphi$.

(ii) If Γ is consistent then the set of maximal models of Γ is equal to the set of maximal $\models^*$-models of Γ.

Proof. (i): Assume that $\pi' \models \varphi$. Since $\pi \models^* \Gamma$ we have $\pi \models^* \varphi$, and so $\pi \circ \pi' \models \varphi$, since φ is strongly compositional. Because $\pi \circ \pi' \sqsubseteq \pi$ and $\pi \circ \pi' \models^* \varphi$ and π is a maximal $\models^*$-model of Γ, $\pi \circ \pi' = \pi$ and thus $\pi \models \varphi$. Reversing the roles of π and π' in the argument, we have $\pi \models \varphi \iff \pi' \models \varphi$.

(ii): Let Γ be consistent. We first prove that any maximal model π of Γ is also a maximal $\models^*$-model of Γ. Let π be a maximal model of Γ and assume that

π is not a maximal $\models^*$-model of Γ. Then there exists a model π' extending π, $\pi' \sqsupset \pi$, that is a $\models^*$-model of Γ. Let π' be a maximal model that satisfies this, i.e., there exists no extension of π' that is a $\models^*$-model of Γ (which exists, since $\mathcal{G}$ is assumed to be finite). Then π' is a maximal $\models^*$-model of Γ. Since Γ is strongly compositional, by Lemma 4.13, $\pi' \circ \pi \models \Gamma$, and thus by Lemma 4.10, $\pi' \circ \pi \models^* \Gamma$. But since $\pi' \circ \pi \sqsupseteq \pi'$ and π' was chosen to be a maximal $\models^*$-model of Γ, $\pi' \circ \pi = \pi'$. Thus, $\pi' \models \Gamma$ and $\pi' \sqsupset \pi$. This is a contradiction to the maximality of π. Hence, π is a maximal $\models^*$-model of Γ.

We now prove that any maximal $\models^*$-model of Γ is a maximal model of Γ. Let π be some maximal $\models^*$-model of Γ. Then by Proposition 4.12 there exists a model π' with $\pi' \sqsupseteq \pi$ that is a maximal model of Γ. Then $\pi' \models^* \Gamma$ by Lemma 4.10, and $\pi' \sqsupseteq \pi$. Since π is a maximal $\models^*$-model, $\pi' = \pi$, and thus π is a maximal model of Γ. □

The following corollary follows immediately from the second part of Theorem 4.1 and together with the next corollary forms the base of our algorithmic approach to solve the Consistency Problem for strongly compositional statements discussed in the next section.

Corollary 4.17. *If $\Gamma \subseteq \mathcal{L}$ is consistent and strongly compositional then every maximal $\models^*$-model of Γ satisfies Γ.*

Proof. By Theorem 4.1, if Γ is consistent and strongly compositional then every maximal $\models^*$-model of Γ is a maximal model of Γ and thus satisfies Γ. □

Reformulating this result implies the following corollary.

Corollary 4.18. *Let π be any maximal $\models^*$-model of strongly compositional $\Gamma \subseteq \mathcal{L}$. Then Γ is consistent if and only if $\pi \models \Gamma$.*

4.2.4 An Algorithmic Approach for the Consistency Problem with Strongly Compositional Statements

We now aim at formulating an algorithm to solve the Consistency Problem for strongly compoositional preference statements. Corollary 4.18 shows that we can test consistency of strongly compositional Γ, by finding any maximal $\models^*$-model π of it, and checking if π satisfies Γ. In this section we show how a maximal $\models^*$-model of Γ can be constructed iteratively by minimal extensions,

starting from a minimum model $\pi_{min\mathcal{G}}$ or a minimal model. To this end, we first define minimum models and minimal extensions and show their connection to consistency and inference. After formally describing the algorithm, we discuss how the algorithm can be extended to work for minimal models.

4.2.4.1 Minimum Model and Minimal Extensions

Let us define minimum models, which are the initial models in our greedy approach to solve the Consistency Problem for strongly compositional preference statements.

Definition 4.7: Minimum Models

A minimum model in $\mathcal{G}$ is a preference model $\pi_{min\mathcal{G}} \in \mathcal{G}$ such that any other model $\pi \in \mathcal{G}$ is an extension $\pi \sqsupseteq \pi_{min\mathcal{G}}$.

Note that a minimum model of $\mathcal{G}$ is not guaranteed to exist. Also, by definition, for two minimum models $\pi_{min\mathcal{G}}$ and $\pi'_{min\mathcal{G}}$, $\pi'_{min\mathcal{G}} \sqsupseteq \pi_{min\mathcal{G}}$ and $\pi_{min\mathcal{G}} \sqsupseteq \pi'_{min\mathcal{G}}$. Then by Proposition 4.2, $\pi_{min\mathcal{G}} = \pi'_{min\mathcal{G}}$. Thus, if there exists a minimum model, then it is unique. Note also that the definition of a minimum model, as for maximal models, depends on a composition operator $\circ$ and an associated extension relation $\sqsupseteq$. For the examples for composition operators $\circ_{\mathcal{H}(1)}$, $\circ_{\mathscr{L}}$, $\circ_{\mathcal{P}(1)}$ and $\circ_{\mathscr{P}}$ for lexicographic and singleton Pareto models, the minimum model is the empty model. If a minimum model exists in $\mathcal{G}$, we can check if $\pi_{min\mathcal{G}} \models^* \Gamma$. The following lemma shows that this is equivalent to Γ being $\models^*$-consistent, i.e., there exists some $\pi \in \mathcal{G}$ with $\pi \models^* \Gamma$, which is a very weak property because it just requires that each element of Γ is (individually) consistent.

Lemma 4.19. *Let $\Gamma \subseteq \mathcal{L}$ and suppose that $\mathcal{G}$ has a minimum model $\pi_{min\mathcal{G}}$. Γ is $\models^*$-consistent if and only if, for each $\varphi \in \Gamma$, φ is consistent. This also holds if and only if $\pi_{min\mathcal{G}} \models^* \Gamma$.*

Proof. Suppose first that Γ is $\models^*$-consistent, and that $\pi \models^* \Gamma$. Then, for any $\varphi \in \Gamma$ there exists some $\pi' \in \mathcal{G}$ with $\pi' \sqsupseteq \pi$ and $\pi' \models \varphi$, which implies that every $\varphi \in \Gamma$ is consistent.

Now, assume that all $\varphi \in \Gamma$ are consistent. Then for all $\varphi \in \Gamma$, there exists some $\pi \in \mathcal{G}$ with $\pi \models \varphi$. By definition of $\pi_{min\mathcal{G}}$, this entails that $\pi_{min\mathcal{G}} \models^* \varphi$ for all $\varphi \in \Gamma$. Thus, $\pi_{min\mathcal{G}} \models^* \Gamma$.

By definition of consistency, $\pi_{min\mathcal{G}} \models^* \Gamma$ implies Γ is $\models^*$-consistent. □

The condition in Corollary 4.18, requires a maximal $\models^*$-model to check consistency of strongly compositional preference statements. In the following we show that for building a maximal $\models^*$-model, we only need to consider finding minimal extensions.

Definition 4.8: Minimal Extensions

We say that π' *minimally extends* π if $\pi' \sqsupset \pi$ and there exists no intermediate model $\pi'' \in \mathcal{G}$, i.e., a model such that $\pi' \sqsupset \pi'' \sqsupset \pi$.

Lemma 4.20. *π is a maximal $\models^*$-model of Γ if and only if $\pi \models^* \Gamma$ and there exists no π' minimally extending π such that $\pi' \models^* \Gamma$.*

Proof. If π is a maximal $\models^*$-model of Γ, then by definition there exists no extension of π that $\models^*$-satisfies Γ. In particular, there exists no minimal extension that $\models^*$-satisfies Γ.

For the converse, let π be a $\models^*$-model of Γ that is not maximal. Then there exists a $\models^*$-model of Γ strictly extending π. Choose such a model $\pi' \in \mathcal{G}$ such that there exists no intermediate $\models^*$-model $\pi'' \in \mathcal{G}$ of Γ with $\pi' \sqsupset \pi'' \sqsupset \pi$. Note that such a π' exists since $\mathcal{G}$ is assumed to be finite. Then by definition, π' is a minimal extension of π such that $\pi' \models^* \Gamma$. □

4.2.4.2 The Algorithm

If there exists a minimum model $\pi_{min\mathcal{G}}$ of Γ then we can, starting with $\pi_{min\mathcal{G}}$, construct a maximal $\models^*$-model π of Γ by iteratively replacing the current model with one minimally extending it and still $\models^*$-satisfying Γ. If there exists no more extension, we have found a maximal $\models^*$-model π of Γ. By Corollary 4.18, we can test if Γ is consistent by checking $\pi \models \Gamma$.

Algorithm 4.1: Consistency for Strongly Compositional Statements Γ

```
Input:    Set of models 𝒢 with minimum model π_min𝒢,
          strongly compositional statements Γ.
Question: Is Γ consistent with respect to 𝒢?
```

```
1. IF ( π_minG |=* Γ ) THEN
2.      π ← π_minG
3.      WHILE ( ∃ min extension π' ⊐ π with π' |=* Γ )
4.              π ← π'
5.      IF ( π |= Γ ) THEN
6.          RETURN "Γ is consistent" and STOP.
7. RETURN "Γ is inconsistent" and STOP.
```

Proposition 4.21. *Algorithm 4.1 is correct.*

Proof. The correctness of Algorithm 4.1 is a direct consequence of Lemmas 4.19 and 4.20, and Corollary 4.18. □

To analyse the runtime, let us define a *satisfaction test* as a test of the form $\pi \models \varphi$ for some $\pi \in \mathcal{G}$ and $\varphi \in \mathcal{L}$; a $\models^*$*-satisfaction test* is a test of the form $\pi \models^* \varphi$.

Assuming that a minimum model of $\mathcal{G}$ is given, we first check if the minimum model is a $\models^*$-model of Γ, which involves a number $|\Gamma|$ of $\models^*$-satisfaction tests using Lemma 4.19. At each iterative step, we have to do at most $|\Gamma|$ $\models^*$-satisfaction tests for every possible minimal extension of the current model. Finally, we have to check that the produced maximal $\models^*$-model of Γ satisfies Γ, which involves $|\Gamma|$ satisfaction tests. This method can be efficient if a minimum model of $\mathcal{G}$ is given or easy to find, the number of minimal extensions is restricted in every step and easy to compute and satisfaction and $\models^*$-satisfaction tests are efficient to compute. We will describe this method and its efficiency in more detail for (fvo and cvo) lexicographic models in Chapter 6 and Chapter 8, hierarchical models in Chapter 7, and find alternative approaches for fvo Pareto models in Chapter 5.

4.2.4.3 Minimal Model

Note that, since existence of a minimum model is not guaranteed, Algorithm 4.1 does not apply to every set of strongly compositional statements Γ. However, one could relax the definition of minimum models and transfer this method.

Definition 4.9: Minimal Models

We say a model $\pi \in \mathcal{G}$ is a *minimal model* of a set of models $\mathcal{G}$ if there exists no model $\pi' \in \mathcal{G}$ such that $\pi \sqsupset \pi'$.

By this definition, a minimal model of $\mathcal{G}$ always exists, however is not guaranteed to be unique. If there exists a unique minimal model it is also the minimum model. This follows immediately from the next proposition which expresses a basic property of minimum elements in partially ordered sets.

Proposition 4.22. *Let $\mathcal{B}$ be the set of minimal models for the finite set of models $\mathcal{G}$. Then for any model $\pi \in \mathcal{G}$ there exists a model $\pi' \in \mathcal{B}$ such that $\pi \sqsupseteq \pi'$.*

Proof. If $\pi \in \mathcal{B}$ then the result trivially follows. Suppose π is not a minimal model. Then there exists a model $\pi_1 \in \mathcal{G}$ such that $\pi \sqsupset \pi_1$. The model π_1 is either a minimal model or there exists another model $\pi_2 \in \mathcal{G}$ such that $\pi_1 \sqsupset \pi_2$. By repeating this argument, there exists a sequence of model extensions $\pi \sqsupset \pi_1 \sqsupset \pi_2 \sqsupset \dots$. Since the set $\mathcal{G}$ is assumed to be finite and $\sqsupset$ is transitive (by Proposition 4.4), such a sequence is finite, i.e., there exists π_k such that $\pi \sqsupset \pi_1 \sqsupset \pi_2 \sqsupset \cdots \sqsupset \pi_k$ and there exists no other model that π_k is an extension of. Then $\pi \sqsupset \pi_k$ by transitivity of $\sqsupset$, and π_k is a minimal model. □

We can generalise the statement of Lemma 4.19 as follows.

Lemma 4.23. *For the set of all minimal models $\mathcal{B}$, a set of preference statements Γ is $\models^*$-consistent if and only if there exists $\pi \in \mathcal{B}$ such that $\pi \models^* \Gamma$.*

Proof. By definition of consistency, if there exists $\pi \in \mathcal{B}$ such that $\pi \models^* \Gamma$, then Γ is $\models^*$-consistent. Assume Γ is $\models^*$-consistent. Then there exists a model $\pi \in \mathcal{G}$ such that $\pi \models^* \Gamma$. By Proposition 4.22, there exists a model $\pi' \in \mathcal{B}$ such that $\pi \sqsupseteq \pi'$ and by Lemma 4.10, $\pi' \models^* \Gamma$. □

By these results, we can construct a maximal $\models^*$-model of Γ iteratively as described before starting with a minimal $\models^*$-model of Γ instead of a minimum $\models^*$-model of Γ.

4.2.5 Decreasing Preference Statements and Sets of Models

Since there exists a promising algorithmic approach to solve the Consistency Problem, as shown in the last section, it can be helpful for any new preference framework with preference models and statements of some types to check for

strong compositionality. In this section we show some results that are useful in proving that certain preference statements are strongly compositional.

The first important property that can help us to show strong compositionality is the one that a statement is *decreasing*.

Definition 4.10: Decreasing Statements

We say that $\varphi \in \mathcal{L}$ is *decreasing* if for all $\pi, \pi' \in \mathcal{G}$ with π' extending π, we have $\pi' \models \varphi \Rightarrow \pi \models \varphi$.

It follows easily from the definitions that if φ is decreasing, then, for all $\pi \in \mathcal{G}$, $\pi \models^* \varphi \iff \pi \models \varphi$. This leads to the following result.

Lemma 4.24. *Let $\varphi \in \mathcal{L}$ be decreasing. Then φ is strongly compositional if and only if φ is compositional.*

Proof. φ is strongly compositional if and only if for all $\pi, \pi' \in \mathcal{G}$ with $\pi \models^* \varphi$ and $\pi' \models \varphi$, $\pi \circ \pi' \models \varphi$. Since φ is decreasing, $\pi \models^* \varphi$ is equivalent to $\pi \models \varphi$. Thus, φ is strongly compositional if and only if for all $\pi, \pi' \in \mathcal{G}$ with $\pi \models \varphi$ and $\pi' \models \varphi$, $\pi \circ \pi' \models \varphi$, which is if and only if φ is compositional. □

Example 4.3

As before, consider flight connections with variables $\mathcal{V} = \{$airline, class, time$\}$ and fixed value orders KLM > LAN, business > economy and day > night. Let the statement φ be given by (KLM, economy, night) $\geq$ (LAN, business, day).

As argued in a previous example, the model (airline, time, class) is a maximal model of φ. The only other maximal model of φ is (airline, class, time). The two maximal models are extending the models (airline, time), (airline, class), (airline) and (), all of which satisfy the statement as well. Thus, (KLM, economy, night) $\geq$ (LAN, business, day) is decreasing.

The statement φ' given by (LAN, business, night) $>$ (LAN, economy, day) is satisfied by the model (airline, class, time) which also is a maximal model of φ' and an extension of the model (airline). However, (airline) does not satisfy φ'. Thus, φ' is not decreasing.

We will show that in general non-strict statements on complete variable assignments in the context of CVO lexicographic models are decreasing, while strict statements are not, see Section 4.3.2.

We now introduce other criteria by which we can check strong compositionality of preference statements. These are connected to properties of sets of preference models. For this, we first define decreasing sets.

Definition 4.11: Decreasing Sets

For $\mathcal{M} \subseteq \mathcal{G}$, we say that $\mathcal{M}$ is *decreasing* if, for any $\pi, \pi' \in \mathcal{G}$ such that π' extends π, we have $\pi' \in \mathcal{M} \Rightarrow \pi \in \mathcal{M}$.

We now define the notion of a set of preference models that contains all models of a statement.

Definition 4.12: Containing Models of Statements

We say that $\mathcal{M}$ *contains all models of* φ if, for all $\pi \in \mathcal{G}$, $\pi \models \varphi \Rightarrow \mathcal{M} \ni \pi$.

The following result is helpful for proving that a preference statement φ is strongly compositional.

Proposition 4.25. *Let $\varphi \in \mathcal{L}$ be consistent and let $\mathcal{M}_\varphi$ be a subset of $\mathcal{G}$. The following two conditions are equivalent:*

(I) $\mathcal{M}_\varphi$ is decreasing and contains all models of φ, and for all $\pi, \pi' \in \mathcal{G}$, if $\pi \in \mathcal{M}_\varphi$ and $\pi' \models \varphi$ then $\pi \circ \pi' \models \varphi$.

(II) φ is strongly compositional, and for all $\pi \in \mathcal{G}$, $\pi \models^* \varphi \iff \pi \in \mathcal{M}_\varphi$.

Proof. (I) $\Rightarrow$ (II): Let $\pi \in \mathcal{M}_\varphi$. Since φ is consistent, there exists π' with $\pi' \models \varphi$, and so (I) implies that $\pi \circ \pi' \models \varphi$. Thus, $\pi \models^* \varphi$, since $\pi \circ \pi' \sqsupseteq \pi$. For proving the converse, let us now assume that $\pi \models^* \varphi$, so there exists $\pi' \in \mathcal{G}$ with $\pi' \sqsupseteq \pi$ and $\pi' \models \varphi$. Thus, $\pi' \in \mathcal{M}_\varphi$, and because $\mathcal{M}_\varphi$ is decreasing, we then have $\pi \in \mathcal{M}_\varphi$. We have shown that for all $\pi \in \mathcal{G}$, $\pi \models^* \varphi \iff \pi \in \mathcal{M}_\varphi$. (I) then also implies that φ is strongly compositional.

(II) $\Rightarrow$ (I): Lemma 4.10(i) implies that $\mathcal{M}_\varphi$ contains all models of φ, and Lemma 4.10(ii) implies that $\mathcal{M}_\varphi$ is decreasing. The fact that φ is strongly compositional then implies (I). □

The last criterion in this subsection by which strong compositionality can be checked, needs the definition of relaxations of statements.

Definition 4.13: Relaxations of Statements

For $\varphi, \varphi' \in \mathcal{L}$, we say that φ' *is a relaxation of* φ if $\varphi \models \varphi'$, i.e., for all $\pi \in \mathcal{G}$, $\pi \models \varphi \Rightarrow \pi \models \varphi'$.

We have the following special case of Proposition 4.25.

Proposition 4.26. *Let $\varphi, \bar{\varphi} \in \mathcal{L}$, and assume that φ is consistent. The following two conditions are equivalent:*

(I) $\bar{\varphi}$ is a decreasing relaxation of φ such that for all $\pi, \pi' \in \mathcal{G}$, if $\pi \models \bar{\varphi}$ and $\pi' \models \varphi$ then $\pi \circ \pi' \models \varphi$.

(II) φ is strongly compositional, and for all $\pi \in \mathcal{G}$, $\pi \models^* \varphi \iff \pi \models \bar{\varphi}$.

Proof. Define $\mathcal{M}_\varphi$ to be all $\pi \in \mathcal{G}$ such that $\pi \models \bar{\varphi}$. Note that (I) holds if and only if $\mathcal{M}_\varphi$ is decreasing and contains all models of φ. Also, if $\pi \in \mathcal{M}_\varphi$ and $\pi' \models \varphi$ then $\pi \circ \pi' \models \varphi$. Proposition 4.25 implies that (I) $\iff$ (II). □

4.3 Examples for Specific Model Types

In the following, we will analyse preference inference for preference languages under (strong) compositionality as in the previous sections. However, as outlined in Chapter 3, many preference models typically involve a set of variables by which alternatives can be described. We consider some generalised results for composition operators and preference models that can be mapped to a set of variables in a specific way. More detailed results are given for the cases of cvo lexicographic models and cvo Pareto models. Here, the results about cvo lexicographic models are based on [WG17].

4.3.1 Models and Composition Based on Unions of Variables

In this subsection we consider a set of preference models $\mathcal{G}$ that can be mapped to variables $\mathcal{V}$. That is, every $\pi \in \mathcal{G}$ can be associated with a set of variables

$V_\pi \subseteq \mathcal{V}$. We assume that $\circ$ is a composition operator on $\mathcal{G}$, i.e., $\circ$ satisfies associativity, idempotence and asymmetry.

Let us first formally define a variable mapping that maps preference models to sets of variables.

Definition 4.14: Variable Mappings

Let $\circ$ be a composition operator on a set of preference models $\mathcal{G}$, and let $\mathcal{V}$ be a set of variables. Then $V : \mathcal{G} \longrightarrow 2^{\mathcal{V}}$, where $2^{\mathcal{V}}$ is the power set of variables, is a *variable mapping* if the following three properties hold. Here, we abbreviate $V(\pi)$ to V_π for models $\pi \in \mathcal{G}$.

(i) The composition $\pi \circ \pi'$ of models $\pi, \pi' \in \mathcal{G}$ is mapped to the union of variables V_π and $V_{\pi'}$, i.e., $V_{\pi\circ\pi'} = V_\pi \cup V_{\pi'}$.

(ii) For $\pi \in \mathcal{G}$ with $V_\pi \neq \emptyset$, there exists a variable set $V \subsetneq V_\pi$ and a model $\pi' \in \mathcal{G}$ with $V_{\pi'} = V$ and $\pi \sqsupset \pi'$.

(iii) If $V_{\pi'} \subseteq V_\pi$ for models π and π', then $\pi = \pi \circ \pi'$.

For the remainder of this section, $V : \mathcal{G} \longrightarrow 2^{\mathcal{V}}$ will always denote a variable mapping.

As described in Section 4.2.4, we can test consistency of strongly compositional Γ, by finding any maximal $\models^*$-model π of it, and checking if π satisfies Γ. Starting with any consistent minimal model of $\mathcal{G}$ we grow a maximal $\models^*$-model of Γ, by (iteratively) replacing the model with one minimally extending it and still $\models^*$-satisfying Γ, if such a model exists. Otherwise, we have a maximal $\models^*$-model π of Γ. We can prove that for a variable mapping $V : \mathcal{G} \longrightarrow 2^{\mathcal{V}}$, π' *minimally extending* π implies $|V_{\pi'}| = |V_\pi| + 1$, i.e., π' involves one more variable than π. Also, we can show that under properties (i)–(iii) for variable mappings, all maximal $\models^*$-models of strongly compositional preference statements Γ get mapped to the same set of variables. This implies that for building a maximal $\models^*$-model, we only need to consider adding one variable at a time to the associated variable sets. In the following, we prove the necessary results starting with the property that two maximal models are mapped to the same variable set.

Proposition 4.27. *If Γ is consistent and strongly compositional and π and π' are two maximal $\models^*$-models of Γ, then $V_\pi = V_{\pi'}$.*

Proof. Let π and π' be two maximal $\models^*$-models of Γ. Suppose $V_\pi \neq V_{\pi'}$ and w.l.o.g. assume $\emptyset \neq V_{\pi'} \setminus V_\pi$. Consider the composition $\pi \circ \pi'$. Since Γ is strongly compositional, by Lemma 4.15, $\pi \circ \pi' \models^* \Gamma$. Also, $\pi \circ \pi' \sqsupseteq \pi$. Since $V_{\pi\circ\pi'} \neq V_\pi$ and V is a mapping, $\pi \circ \pi' \neq \pi$, and thus $\pi \circ \pi' \sqsupset \pi$. This is a contradiction to the $\models^*$-maximality of π. Thus $\emptyset = V_{\pi'} \setminus V_\pi$. Analogously, we can prove $\emptyset = V_\pi \setminus V_{\pi'}$. Hence $V_\pi = V_{\pi'}$. □

We now prove that the variable set of a minimal extension of a model contains exactly one more variable.

Proposition 4.28. *If π' is a minimal extension of π, then $|V_{\pi'}| = |V_\pi| + 1$.*

Proof. Suppose π' is a minimal extension of π (not equalling π) and $\pi' = \pi \circ \pi_1$. By property (i) of variable mappings, $V_{\pi'} = V_{\pi\circ\pi_1} = V_\pi \cup V_{\pi_1}$, thus, $V_{\pi'} \supseteq V_\pi$ and $|V_{\pi'}| \geq |V_\pi|$. Now assume $|V_{\pi'} \setminus V_\pi| = |V_{\pi_1} \setminus V_\pi| \geq 2$. By property (ii) of variable mappings, there exists $V = \{X\}$ for some variable $X \in V_{\pi_1} \setminus V_\pi$ and a model π'' such that $V_{\pi''} = V$ and $\pi_1 \sqsupset \pi''$. Then $\pi \circ \pi'' \sqsupseteq \pi$, and $\pi \circ \pi'' \neq \pi$ since $V_{\pi\circ\pi''} = V_\pi \cup V_{\pi''} \neq V_\pi$. Thus, $\pi \circ \pi'' \sqsupset \pi$. Also, by Lemma 4.1, $\pi \circ \pi_1 \sqsupseteq \pi \circ \pi''$. Furthermore, since $V_{\pi\circ\pi_1} \neq V_{\pi\circ\pi''}$ and V is a mapping, $\pi \circ \pi_1 \sqsupset \pi \circ \pi''$. Hence $\pi' \sqsupset \pi \circ \pi'' \sqsupset \pi$ which is a contradiction to the minimality of the extension π' of π. Thus, $|V_{\pi'} \setminus V_\pi| \leq 1$, i.e., $|V_{\pi'}| \leq |V_\pi| + 1$.

Now assume, $|V_{\pi'}| = |V_\pi|$. Then $V_{\pi'} = V_\pi$ and thus $V_{\pi_1} \subseteq V_\pi$. By property (iii) of variable mappings, $\pi = \pi \circ \pi_1$, i.e., $\pi = \pi'$ which is a contradiction to $\pi' \sqsupset \pi$. Thus, $|V_{\pi'}| \neq |V_\pi|$, and hence $|V_{\pi'}| = |V_\pi| + 1$. □

The next proposition extends the result of Proposition 4.27 and specifies relations between variable sets of Γ-satisfying models further. It states that all maximal models are mapped to the same variables and that all other models are mapped to subsets of these variables.

Proposition 4.29. *Assume that $\Gamma \subseteq \mathcal{L}$ is consistent and compositional. Let π, π' be models of Γ with π a maximal model of Γ. Then, $V_\pi \supseteq V_{\pi'}$. Also, $V_\pi = V_{\pi'}$ if and only if π' is a maximal model of Γ.*

Proof. Let π be a maximal model of Γ and let π' be any model of Γ. Compositionality of Γ implies that $\pi \circ \pi' \models \Gamma$, and thus, $\pi \circ \pi' = \pi$, since π is a maximal model of Γ and $\pi \circ \pi' \sqsupseteq \pi$. Using property (i) of variable mappings, $V_{\pi\circ\pi'} = V_\pi \cup V_{\pi'} = V_\pi$, and hence, $V_{\pi'} \subseteq V_\pi$. If π' is a maximal model of Γ then

the same argument implies that $V_{\pi'} \supseteq V_\pi$ and thus $V_{\pi'} = V_\pi$. Since the models π and π' are arbitrarily chosen, all maximal models contain the same variables.

Let π'' be a model of Γ with the same variable set as the maximal model π of Γ, i.e., $V_{\pi''} = V_\pi$. Assume π'' is not a maximal model. Then, by Proposition 4.11, there exists an extension $\bar{\pi} \sqsupset \pi''$ such that $\bar{\pi}$ is a maximal model of Γ. By Proposition 4.28, $|V_{\pi''}| < |V_{\bar{\pi}}|$ and thus $V_\pi = V_{\pi''} \subsetneq V_{\bar{\pi}}$. This is a contradiction, since both π and $\bar{\pi}$ are maximal models of Γ and by the first part $V_\pi = V_{\bar{\pi}}$. Thus, every model that includes all variables of a maximal model is a maximal model itself. □

Proposition 4.29 justifies the next definition, and implies the following lemma.

Definition 4.15: Maximal Models Variable Set V^Γ

For consistent and strongly compositional $\Gamma \subseteq \mathcal{L}$, we denote the set of variables appearing in any maximal model of Γ by V^Γ, i.e., $V^\Gamma = V_\pi$, for any maximal model π of Γ.

We can exploit the fact that maximal models are mapped to the same variables and define a notion of max-model inference, which is characterised by the lemma below.

Definition 4.16: Maximal Model Inference

We define $\pi \models^{max} \Gamma$ if π is a maximal model of Γ. We also define $\Gamma \models^{max} \varphi$ if $\pi \models \varphi$ for every maximal model π of Γ.

Lemma 4.30. *Let $\Gamma \cup \{\varphi, \neg\varphi\} \subseteq \mathcal{L}$ and suppose that $\Gamma \cup \{\neg\varphi\}$ is compositional. If $\Gamma \models \varphi$ then $\Gamma \models^{max} \varphi$. Now suppose that $\Gamma \not\models \varphi$ and let π be any maximal model of $\Gamma \cup \{\neg\varphi\}$. Then, $\Gamma \models^{max} \varphi \Rightarrow \pi \not\models^{max} \Gamma$.*

Proof. The definitions immediately imply that if $\Gamma \models \varphi$ then $\Gamma \models^{max} \varphi$. Now assume that $\Gamma \not\models \varphi$, and so $\Gamma \cup \{\neg\varphi\}$ is consistent, and let π be any maximal model of $\Gamma \cup \{\neg\varphi\}$. Suppose that $\Gamma \models^{max} \varphi$. The fact that $\pi \not\models \varphi$ implies that $\pi \not\models^{max} \Gamma$. □

We can extend this result in the following way for strongly compositional statements.

Lemma 4.31. *Let $\Gamma \cup \{\varphi, \neg\varphi\} \subseteq \mathcal{L}$ and suppose that $\Gamma \cup \{\neg\varphi\}$ is strongly compositional. Suppose $\Gamma \not\models \varphi$ and let π be any maximal model of $\Gamma \cup \{\neg\varphi\}$. Then, $\Gamma \models^{max} \varphi \iff \pi \not\models^{max} \Gamma$, which holds if and only if $V^{\Gamma\cup\{\neg\varphi\}} \neq V^{\Gamma}$.*

Proof. Assume that $\Gamma \not\models \varphi$, and so $\Gamma \cup \{\neg\varphi\}$ is consistent, and let π be any maximal model of $\Gamma \cup \{\neg\varphi\}$. Suppose that $\Gamma \models^{max} \varphi$. The fact that $\pi \not\models \varphi$ but all maximal models of Γ satisfy φ implies that $\pi \not\models^{max} \Gamma$. Conversely, assume that $\Gamma \not\models^{max} \varphi$ so there exists $\pi' \in \mathcal{G}$ such that $\pi' \models^{max} \Gamma$ and $\pi' \models \neg\varphi$, and thus $\pi' \models \Gamma \cup \{\neg\varphi\}$. Since $\pi' \models \Gamma \cup \{\neg\varphi\}$ and $\pi \models^{max} \Gamma \cup \{\neg\varphi\}$, by Proposition 4.29 we have $V_{\pi'} \subseteq V_{\pi}$. Since $\pi \models \Gamma$ and $\pi' \models^{max} \Gamma$, by Proposition 4.29 we also have $V_{\pi} \subseteq V_{\pi'}$, and thus $V_{\pi'} = V_{\pi}$. The second part of Proposition 4.29 then implies that $\pi \models^{max} \Gamma$.

We have $V^{\Gamma\cup\{\neg\varphi\}} = V_{\pi} \subseteq V^{\Gamma}$. Proposition 4.29 implies that $\pi \models^{max} \Gamma$ if and only if $V_{\pi} = V^{\Gamma}$, which is if and only if $V^{\Gamma\cup\{\neg\varphi\}} = V^{\Gamma}$. □

4.3.2 CVO Lexicographic Models

In this section, we consider the composition operator $\circ_{\mathscr{L}}$ for cvo lexicographic models $\mathscr{L}$ (see Lemma 4.7) so that all results from Sections 4.1 and 4.2 can be applied. We develop additional results for inference and strong compositionality based on cvo lexicographic models together with $\circ_{\mathscr{L}}$ that cannot be generalised trivially. For simplicity of notation, we will abbreviate $\circ_{\mathscr{L}}$ to $\circ$.

Recall that cvo lexicographic models $\mathscr{L}$ are defined over a set of variables $\mathcal{V}$ by which the alternatives can be described, i.e., $\mathcal{A} = \underline{\mathcal{V}}$. $\mathscr{L}$ includes all sequences of the form $(Y_1, \geq_{Y_1}), \ldots, (Y_k, \geq_{Y_k})$, where Y_i, $i = 1, \ldots, k$, are different variables in $\mathcal{V}$, and each $\geq_{Y_i}$ is a total order on the domain $\underline{Y_i}$.

The associated relation $\succcurlyeq_{\pi} \subseteq \mathcal{A} \times \mathcal{A}$ for a cvo lexicographic model $\pi \in \mathscr{L}$ with $\pi = (Y_1, \geq_{Y_1}), \ldots, (Y_k, \geq_{Y_k})$ is defined as follows: for alternatives α and β, $\alpha \succcurlyeq_{\pi} \beta$ ($\pi \models \alpha \geq \beta$) if and only if either (i) for all $i = 1, \ldots, k$, $\alpha(Y_i) = \beta(Y_i)$; or (ii) there exists $i \in \{1, \ldots, k\}$ such that for all $j < i$, $\alpha(Y_j) = \beta(Y_j)$ and $\alpha(Y_i) >_{Y_i} \beta(Y_i)$ (i.e., $\alpha(Y_i) \geq_{Y_i} \beta(Y_i)$ and $\alpha(Y_i) \neq \beta(Y_i)$). Thus $\succcurlyeq_{\pi}$ is a total preorder on $\mathcal{A}$, which is a total order if $k = |\mathcal{V}|$.

The corresponding strict relation $\succ_{\pi}$ is given by $\alpha \succ_{\pi} \beta$ ($\pi \models \alpha > \beta$) if and only if there exists $i \in \{1, \ldots, k\}$ such that $\alpha(Y_i) >_{Y_i} \beta(Y_i)$ and for all $j < i$, $\alpha(Y_j) = \beta(Y_j)$. The corresponding equivalence relation $\equiv_{\pi}$ is given by $\alpha \equiv_{\pi} \beta$

($\pi \models \alpha \equiv \beta$) if and only if for all $i = 1, \ldots, k$, $\alpha(Y_i) = \beta(Y_i)$. Thus, $\alpha \equiv_\pi \beta$ if and only if $\alpha(V_\pi) = \beta(V_\pi)$, where $V_\pi = \{Y_1, \ldots, Y_k\}$ is set of the variables involved in π.

We say π satisfies $\alpha \geq \beta$, denoted $\pi \models \alpha \geq \beta$, if $\alpha \succcurlyeq_\pi \beta$. Similarly, π satisfies $\alpha > \beta$, denoted $\pi \models \alpha \geq \beta$, if $\alpha \succ_\pi \beta$. We have that $\pi \models \alpha \geq \beta \iff \pi \not\models \beta > \alpha$, i.e., $\pi \models \neg(\beta > \alpha)$. So $\alpha \geq \beta$ and $\neg(\beta > \alpha)$ are equivalent preference statements, in that they are satisfied by exactly the same set of lex models. Similarly, $\alpha > \beta$ and $\neg(\beta \geq \alpha)$ are equivalent preference statements.

As shown in Proposition 4.7 in the beginning of the chapter, there exists an operator satisfying properties 1)—3) of compositions that is defined as follows. Let $\pi = (Y_1, \geq_{Y_1}), \ldots, (Y_k, \geq_{Y_k})$, and $\pi' = (Z_1, \geq_{Z_1}), \ldots, (Z_l, \geq_{Z_l})$ be two cvo lexicographic models. Let π'' be the sequence π' but where pairs $(Z_i, \geq_{Z_i})$ are omitted if $Z_i \in V_\pi$. Define lex model $\pi \circ \pi'$ to be π followed by π''. Then, any lex model π can be mapped to the set $\sigma(\pi) = V_\pi \subseteq \mathcal{V}$ of variables involved and $V_{\pi\circ\pi'} = V_\pi \cup V_{\pi'}$. For any initial sequence of variables V in π, we can construct a model π' that involves the variables $V_{\pi'} = V$ in exactly the order as in π, such that $\pi \sqsupset \pi'$. Furthermore, if $V_{\pi'} \subseteq V_\pi$ for models π and π', then $\pi = \pi \circ \pi'$. Thus, V_π (i.e., $\sigma(\pi)$) is a variable mapping and all results from Section 4.3.1 (Propositions 4.27 — 4.29 and Lemma 4.31) hold.

The next proposition shows that if two alternatives are equivalent in one maximal cvo lexicographic model of Γ then they are equivalent in all models of Γ.

Proposition 4.32. *Assume that $\Gamma \subseteq \mathcal{L}$ is consistent and compositional. Let $\pi, \pi' \in \mathscr{L}$ be models of Γ and let π be a maximal model of Γ. If, for $\alpha, \beta \in \mathcal{A}$, we have $\alpha \equiv_\pi \beta$ then we have $\alpha \equiv_{\pi'} \beta$, and in fact $\Gamma \models \alpha \equiv \beta$.*

Proof. By Proposition 4.29, $V_{\pi'} \subseteq V_\pi$. Assume that $\alpha \equiv_\pi \beta$. Then $\alpha(V_\pi) = \beta(V_\pi)$, and so $\alpha(V_{\pi'}) = \beta(V_{\pi'})$ and $\pi' \models \alpha \equiv \beta$. Since π' is arbitrary, we have $\Gamma \models \alpha \equiv \beta$. □

4.3.2.1 Statements on Complete Variable Assignments

Let us now consider simple preference statements that are direct comparisons between alternatives. In the following, we will show that strict and non-strict statements are strongly compositional, making use of some of the criteria that were developed in Section 4.2.5.

We have the following basic monotonicity property for strict statements.

Lemma 4.33. *For $\pi, \pi' \in \mathscr{L}$, suppose that π' extends π. If $\alpha \succ_\pi \beta$ then $\alpha \succ_{\pi'} \beta$.*

Proof. If $\alpha \succ_\pi \beta$ then α and β differ on some variable in π, and α is better than β on the first such variable; thus the same holds for π', since π' extends π, and so $\alpha \succ_{\pi'} \beta$ holds. □

We can now show that non-strict statements are decreasing.

Lemma 4.34. *$\alpha \geq \beta$ is decreasing.*

Proof. Let $\pi, \pi' \in \mathscr{L}$ such that π' extends π. By Lemma 4.33, if $\beta \succ_\pi \alpha$ then $\beta \succ_{\pi'} \alpha$, which, using the fact that $\succ_\pi$ and $\succ_{\pi'}$ are both weak orders, is equivalent to: if $\alpha \not\succ_\pi \beta$ then $\alpha \not\succ_{\pi'} \beta$. This implies, if $\alpha \succcurlyeq_{\pi'} \beta$ then $\alpha \succcurlyeq_\pi \beta$. Thus, $\alpha \geq \beta$ is decreasing. □

The following lemma shows results of satisfaction for strict and non-strict preference statements under the composition of preference models. These results will be used in order to show strong compositionality of strict and non-strict preference statements.

Lemma 4.35. *Let $\pi, \pi' \in \mathscr{L}$. If $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succ_{\pi'} \beta$, then $\alpha \succ_{\pi \circ \pi'} \beta$ and $\alpha \succ_{\pi' \circ \pi} \beta$. If $\alpha \equiv_\pi \beta$, then $\alpha \succcurlyeq_{\pi'} \beta \iff \alpha \succcurlyeq_{\pi \circ \pi'} \beta$; and $\alpha \succ_{\pi'} \beta \iff \alpha \succ_{\pi \circ \pi'} \beta$.*

Proof. Since $\pi' \circ \pi$ extends π', we have, by Lemma 4.33, that (i) $\alpha \succ_{\pi'} \beta$ implies $\alpha \succ_{\pi' \circ \pi} \beta$. Similarly, (ii) if $\alpha \succ_\pi \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$, and thus also $\alpha \succcurlyeq_{\pi \circ \pi'} \beta$. We also have (iii) if $\alpha \equiv_\pi \beta$ and $\alpha \succ_{\pi'} \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$. This is because α and β differ on some variable in π', and the first such variable pair $(X, \geq)$ that appears in the sequence of π' satisfies $\alpha(X) > \beta(X)$. Since $\alpha \equiv_\pi \beta$, X is also the earliest variable in $\pi \circ \pi'$ that α and β differ on, so $\alpha \succ_{\pi \circ \pi'} \beta$.

Suppose that $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succ_{\pi'} \beta$. Then, by (i), $\alpha \succ_{\pi' \circ \pi} \beta$. If $\alpha \succ_\pi \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$ by (ii), and if $\alpha \equiv_\pi \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$ follows by (iii). Hence, if $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succ_{\pi'} \beta$ then $\alpha \succ_{\pi' \circ \pi} \beta$ and $\alpha \succ_{\pi \circ \pi'} \beta$.

Assume now that $\alpha \equiv_\pi \beta$, so that $\alpha \succcurlyeq_\pi \beta$ and $\beta \succcurlyeq_\pi \alpha$. By the first part, if $\alpha \succ_{\pi'} \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$. If $\alpha \equiv_{\pi'} \beta$ then $\alpha(V_{\pi'}) = \beta(V_{\pi'})$ and thus $\alpha(V_{\pi \circ \pi'}) = \beta(V_{\pi \circ \pi'})$, i.e., $\alpha \equiv_{\pi \circ \pi'} \beta$. Thus we have $\alpha \succcurlyeq_{\pi'} \beta \Rightarrow \alpha \succcurlyeq_{\pi \circ \pi'} \beta$, and $\alpha \succ_{\pi'} \beta \Rightarrow \alpha \succ_{\pi \circ \pi'} \beta$. Since $\beta \succcurlyeq_\pi \alpha$, by the first part we also have if $\alpha \not\succ_{\pi'} \beta$, i.e., $\beta \succ_{\pi'} \alpha$, then $\beta \succ_{\pi \circ \pi'} \alpha$, i.e., $\alpha \not\succ_{\pi \circ \pi'} \beta$. Similarly we have $\alpha \not\prec_{\pi'} \beta$ implying $\alpha \not\prec_{\pi \circ \pi'} \beta$. □

Using the previous lemma and and properties of weak orders we can prove that non-strict statements are compositional. Since, by Lemma 4.34 non-strict statements are decreasing, this implies strong compositionality by Lemma 4.24.

Proposition 4.36. *$\alpha \geq \beta$ is strongly compositional.*

Proof. We first prove compositionality of $\alpha \geq \beta$. Suppose that $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succcurlyeq_{\pi'} \beta$. If $\alpha \succ_\pi \beta$ or $\alpha \succ_{\pi'} \beta$ then, by the first part of Lemma 4.35, we have $\alpha \succ_{\pi\circ\pi'} \beta$, and thus also $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$. It remains to deal with the case where $\alpha \equiv_\pi \beta$ and $\alpha \equiv_{\pi'} \beta$. Then $\alpha \succcurlyeq_{\pi'} \beta$ and by the second part of Lemma 4.35, $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$. Thus, $\alpha \geq \beta$ is compositional. By Lemma 4.34, $\alpha \geq \beta$ is decreasing and thus, Lemma 4.24 implies that $\alpha \geq \beta$ is strongly compositional. □

The following proposition shows the relation between ($\models^*$-) inference of strict and non-strict statements.

Proposition 4.37. *For any alternatives $\alpha, \beta \in \mathcal{A}$ and for $\pi \in \mathscr{L}$, if $\alpha \neq \beta$, $\pi \models^* \alpha > \beta \iff \pi \models \alpha \geq \beta \iff \pi \models^* \alpha \geq \beta$.*

Proof. Suppose $\alpha \neq \beta$. Thus, if there exists a model $\pi' \models \alpha > \beta$, then π' is not the empty model and there exists a variable in π' on which α and β differ. Let Y be the first such variable. Then since any model $\pi \sqsubseteq \pi'$ consists of an initial sequence of variables in π', either Y is included in π and thus $\pi \models \alpha > \beta$ or Y is not included and $\pi \models \alpha = \beta$. Thus, $\pi \models^* \alpha > \beta$ implies $\pi \models \alpha \geq \beta$. Also, if $\pi \models \alpha \geq \beta$ and there exists a variable Y with $\alpha(Y) \neq \beta(Y)$, then $\pi \circ (Y, \geq_Y) \models \alpha > \beta$ for $\alpha(Y) >_Y \beta(Y)$. Hence, $\pi \models^* \alpha > \beta$.

$\pi \models \alpha \geq \beta$ by definition implies $\pi \models^* \alpha \geq \beta$. Conversely, let $\pi \models^* \alpha \geq \beta$. Then there exists a model $\pi' \sqsupseteq \pi$ with $\pi' \models \alpha \geq \beta$. By Lemma 4.34, $\alpha \geq \beta$ is decreasing and thus $\pi \models \alpha \geq \beta$. □

The strong compositionality of strict preference statements easily follows.

Proposition 4.38. *$\alpha > \beta$ is strongly compositional.*

Proof. The first part of Lemma 4.35 together with Proposition 4.37 implies that $\alpha > \beta$ is strongly compositional. □

4.3.2.2 Statements $\varphi^{\mathcal{R}}$, Strict Versions and Negations

After showing that strict and non-strict comparisons of alternatives are strongly compositional preference statements, we will now consider conjunctions of these statements $\varphi^{\mathcal{R}}$ and their negations. Furthermore, we will define strict versions of preference statements. Again, we analyse the strong compositionality of the considered preference statements using the criteria developed in Section 4.2.5.

Let $\mathcal{R}$ be a subset of $\mathcal{A} \times \mathcal{A}$, and let $\varphi^{\mathcal{R}}$ be a statement satisfying: $\pi \models \varphi^{\mathcal{R}}$ if and only if $\succcurlyeq_\pi \supseteq \mathcal{R}$. If π' extends π then $\succcurlyeq_\pi \supseteq \succcurlyeq_{\pi'}$ since non-strict statements are decreasing by Lemma 4.34, which implies that $\varphi^{\mathcal{R}}$ is decreasing. For any cvo lexicographic model π we have $\pi \models \varphi^{\mathcal{R}}$ if and only if for all $(\alpha, \beta) \in \mathcal{R}$, $\pi \models \alpha \geq \beta$, which implies that $\varphi^{\mathcal{R}}$ is strongly compositional, by Proposition 4.36 and Lemma 4.14. We therefore have:

Proposition 4.39. *For any $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, the preference statement $\varphi^{\mathcal{R}}$ is strongly compositional, and for any $\pi \in \mathscr{L}$ we have $\pi \models^* \varphi^{\mathcal{R}} \iff \pi \models \varphi^{\mathcal{R}}$*

We define a model $\pi \in \mathscr{L}$ to satisfies $\neg\varphi^{\mathcal{R}}$, i.e., $\pi \models \neg\varphi^{\mathcal{R}}$, if $\pi \not\models \varphi^{\mathcal{R}}$. Consider $\pi \in \mathscr{L}$ with $\pi \not\models \varphi^{\mathcal{R}}$ and consider any $\pi' \in \mathscr{L}$. Then, $\pi \circ \pi'$ extends or equals π, which implies that $\pi \circ \pi' \not\models \varphi^{\mathcal{R}}$, since $\varphi^{\mathcal{R}}$ is decreasing. Thus, we have:

Proposition 4.40. *Preference statement $\neg\varphi^{\mathcal{R}}$ is compositional for any $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$.*

The next lemma shows that for some non-strict statements and $\varphi^{\mathcal{R}}$ satisfaction under one model is equivalent to satisfaction for an extension.

Lemma 4.41. *Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, and let $\pi, \pi' \in \mathscr{L}$ be such that π' extends π. Suppose that for all $(\alpha, \beta) \in \mathcal{R}$ there exists $X \in V_\pi$ such that $\alpha(X) \neq \beta(X)$. Then, for any $(\alpha, \beta) \in \mathcal{R}$, $\alpha \succcurlyeq_\pi \beta \iff \alpha \succcurlyeq_{\pi'} \beta$. Also, $\pi \models \varphi^{\mathcal{R}} \iff \pi' \models \varphi^{\mathcal{R}}$.*

Proof. Consider any $(\alpha, \beta) \in \mathcal{R}$. Let Y be the earliest variable in V_π such that $\alpha(Y) \neq \beta(Y)$ (this is well-defined, by the hypothesis). Since π' extends π, $\alpha \succcurlyeq_\pi \beta \iff \alpha \succcurlyeq_{\pi'} \beta \iff \alpha(Y) >_Y \beta(Y)$, where $>_Y$ is the strict part of the ordering for Y in π. We then have $\pi \models \varphi^{\mathcal{R}}$, if and only if for all $(\alpha, \beta) \in \mathcal{R}$, $\alpha \succcurlyeq_\pi \beta$, if and only if for all $(\alpha, \beta) \in \mathcal{R}$, $\alpha \succcurlyeq_{\pi'} \beta$, if and only if $\pi' \models \varphi^{\mathcal{R}}$. □

We can also show that the satisfaction of a statement $\varphi^{\mathcal{R}}$ by some model is indifferent under composition with a model that is indifferent about all involved variables.

Lemma 4.42. *Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, and let $\pi, \pi' \in \mathscr{L}$. Suppose that for all $(\alpha, \beta) \in \mathcal{R}$ we have $\alpha(V_\pi) = \beta(V_\pi)$. Then, $\succcurlyeq_{\pi'} \cap \mathcal{R} = \succcurlyeq_{\pi\circ\pi'} \cap \mathcal{R}$, and thus, $\pi' \models \varphi^{\mathcal{R}} \iff \pi \circ \pi' \models \varphi^{\mathcal{R}}$.*

Proof. Consider any $(\alpha, \beta) \in \mathcal{R}$. We have $\alpha \equiv_\pi \beta$. By Lemma 4.35, $\alpha \succcurlyeq_{\pi'} \beta \iff \alpha \succcurlyeq_{\pi\circ\pi'} \beta$. This implies $\succcurlyeq_{\pi'} \cap \mathcal{R} = \succcurlyeq_{\pi\circ\pi'} \cap \mathcal{R}$, and $\pi' \models \varphi^{\mathcal{R}} \iff \pi \circ \pi' \models \varphi^{\mathcal{R}}$. □

Strict Versions:

In the following, we consider strict versions of statements and conditions for their compositionality.

Definition 4.17: Strict Versions of Statements

We say that ψ is a *strict version of* $\varphi^{\mathcal{R}}$, if $\varphi^{\mathcal{R}}$ is a relaxation of ψ (see Definition 4.13), and ψ satisfies the following monotonicity property regarding the strict preferences among $\mathcal{R}$: for all $\pi, \pi' \in \mathscr{L}$, if $\pi \models \psi$ and $\pi' \models \varphi^{\mathcal{R}}$ and $\succ_{\pi'} \supseteq (\succ_\pi \cap \mathcal{R})$ then $\pi' \models \psi$.

There are many strict versions of $\varphi^{\mathcal{R}}$ (unless $\mathcal{R}$ is very small). We give two simple examples ψ_1 and ψ_2 of strict versions of $\varphi^{\mathcal{R}}$ in the following.

Example 4.4

Let ψ_1 be such that $\pi \models \psi_1$ if and only if $\succ_\pi \supseteq \mathcal{R}$. Let ψ_2 be such that $\pi \models \psi_2$ if and only if $\succcurlyeq_\pi \supseteq \mathcal{R}$ and there exists some $(\alpha, \beta) \in \mathcal{R}$ such that $\alpha \succ_\pi \beta$. Then, ψ_1 and ψ_2 are strict versions of $\varphi^{\mathcal{R}}$.

The next proposition shows that strict versions of $\varphi^{\mathcal{R}}$ are strongly compositional.

Proposition 4.43. *Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, and suppose that ψ is a strict version of $\varphi^{\mathcal{R}}$. Then ψ is strongly compositional, and for $\pi \in \mathscr{L}$, $\pi \models^* \psi \iff \pi \models \varphi^{\mathcal{R}}$.*

Proof. We have that $\varphi^{\mathcal{R}}$ is a decreasing relaxation of ψ. Using Proposition 4.26, it is sufficient to prove that for any $\pi, \pi' \in \mathscr{L}$, if $\pi \models \varphi^{\mathcal{R}}$ and $\pi' \models \psi$ then $\pi \circ \pi' \models \psi$. Assume that $\pi \models \varphi^{\mathcal{R}}$ and $\pi' \models \psi$. Since $\varphi^{\mathcal{R}}$ is a relaxation of ψ, we have $\pi' \models \varphi^{\mathcal{R}}$, and thus, $\pi \circ \pi' \models \varphi^{\mathcal{R}}$, since $\varphi^{\mathcal{R}}$ is strongly compositional, by Proposition 4.39. Consider any $(\alpha, \beta) \in \mathcal{R}$ such that $\alpha \succ_{\pi'} \beta$. We also have $\alpha \succcurlyeq_\pi \beta$, because $\pi \models \varphi^{\mathcal{R}}$, and so, $\alpha \succ_{\pi\circ\pi'} \beta$, using Lemma 4.35. We have shown that $\succ_{\pi\circ\pi'} \supseteq (\succ_{\pi'} \cap \mathcal{R})$, which, since ψ is a strict version of $\varphi^{\mathcal{R}}$, implies $\pi \circ \pi' \models \psi$, as required. □

Negated Statements $\neg\varphi^{\mathcal{R}}$:

We define variable projections as in [Wil14] for preference statements, specific model sets and the notion of simultaneous decisiveness, which helps us identify when statements $\neg\varphi^{\mathcal{R}}$ are strongly compositional and satisfied by CVO lexicographic models.

Definition 4.18: Variable Projections

Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, let $Y \in \mathcal{V}$ be a variable, and let $A \subseteq \mathcal{V} - \{Y\}$ be a set of variables not containing Y. Define $\mathcal{R}^{\downarrow Y}$, the projection of $\mathcal{R}$ to Y, to be $\{(\alpha(Y), \beta(Y)) : (\alpha, \beta) \in \mathcal{R}\}$. Also, define, $\mathcal{R}_A^{\downarrow Y}$, the A-restricted projection to Y, to be the set of pairs $(\alpha(Y), \beta(Y))$ such that $(\alpha, \beta) \in \mathcal{R}$ and $\alpha(A) = \beta(A)$.

Note that $\mathcal{R}_A^{\downarrow Y}$ is the projection to Y of pairs that agree on A. Thus, $\mathcal{R}^{\downarrow Y} = \mathcal{R}_\emptyset^{\downarrow Y}$.

From [Wil14] we have (a variation of) the following:

Lemma 4.44. *Consider any* cvo *lexicographic model $\pi \in \mathscr{L}$, written as $(Y_1, \geq_1), \ldots, (Y_k, \geq_k)$, where $Y_i \in \mathcal{V}$ are variables and $\geq_{Y_i}$ are total orders on the variable domains $\underline{Y_i}$. For $i = 1, \ldots, k$, define A_i to be the set of earlier variables than Y_i, i.e., $A_i = \{Y_1, \ldots, Y_{i-1}\}$. Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. Then $\succcurlyeq_\pi \supseteq \mathcal{R}$ if and only if $\geq_i \supseteq \mathcal{R}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$.*

Proof. Consider $(\alpha, \beta) \in \mathcal{R}$.

First, suppose that $(\alpha, \beta) \in \succcurlyeq_\pi$, i.e., $\alpha \succcurlyeq_\pi \beta$. By definition of $\succcurlyeq_\pi$, either (i) for all $i = 1, \ldots, k$, $\alpha(Y_i) = \beta(Y_i)$; or (ii) there exists $i \in \{1, \ldots, k\}$ such that for all $j < i$, we have $\alpha(Y_j) = \beta(Y_j)$ and $\alpha(Y_i) >_{Y_i} \beta(Y_i)$. This is equivalent to either $=_i \supseteq \{(\alpha(Y_i), \beta(Y_i))\} = \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$; or there exists $i \in \{1, \ldots, k\}$ such that for all $j < i$, we have $=_j \supseteq \{(\alpha(Y_j), \beta(Y_j))\} = \{(\alpha, \beta)\}^{\downarrow Y_j}_{A_j}$ and $>_i \supseteq \{(\alpha(Y_i), \beta(Y_i))\} = \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$, i.e., $\{(\alpha, \beta)\}^{\downarrow Y_l}_{A_l} = \emptyset$ for $l > i$. Thus, $\geq_i \supseteq \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$.

Conversely, let $\geq_i \supseteq \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$. Suppose that there exists $i \in \{1, \ldots, k\}$ such that $\alpha(Y_i) <_{Y_i} \beta(Y_i)$. Since $\geq_i \supseteq \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$ and $(\alpha(Y_i), \beta(Y_i)) \in \{(\alpha, \beta)\}^{\downarrow Y_i}$, we must have $\{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i} = \emptyset$, i.e., $\alpha(A_i) \neq \beta(A_i)$. Hence, for every variable Y_i with $\alpha(Y_i) <_{Y_i} \beta(Y_i)$, there exists a variable Y_j with $j < i$ and $\alpha(Y_j) \neq \beta(Y_j)$. Considering the first such variable Y_i in π, implies that there exists a variable Y_l with $l < i$ and $\alpha(Y_j) > \beta(Y_j)$. Thus, either (i) or (ii) holds, i.e. $(\alpha, \beta) \in \succcurlyeq_\pi$.

Since the choice of $(\alpha, \beta) \in \mathcal{R}$ was arbitrary, we have shown that for all $(\alpha, \beta) \in \mathcal{R}$, $(\alpha, \beta) \in \succcurlyeq_\pi$ if and only if $\geq_i \supseteq \{(\alpha, \beta)\}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$. This implies the desired result, $\succcurlyeq_\pi \supseteq \mathcal{R}$ if and only if $\geq_i \supseteq \mathcal{R}^{\downarrow Y_i}_{A_i}$ for all $i = 1, \ldots, k$. □

This result helps us to analyse the structure of $\mathcal{R}^{\downarrow X}$ for some variables X which we will need in the following analysis of the compositionality of $\neg\varphi^{\mathcal{R}}$.

Lemma 4.45. *Let $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, and let $\pi \in \mathscr{L}$. Suppose that X is the first variable in π on which some pair in $\mathcal{R}$ differs, so that there exists $(\alpha, \beta) \in \mathcal{R}$ such that $\alpha(X) \neq \beta(X)$, and this does not hold for any earlier variable in π. If $\pi \models \varphi^{\mathcal{R}}$ then $\mathcal{R}^{\downarrow X}$ is acyclic.*

Proof. Let $\geq_X$ be the non-strict relation for X in π. If $\pi \models \varphi^{\mathcal{R}}$ then, by Lemma 4.44, $\geq_X$ contains $\mathcal{R}^{\downarrow X}$ and thus, since $\geq_X$ is antisymmetric, $\mathcal{R}^{\downarrow X}$ is acyclic. □

Next, we define two sets $\mathcal{M}_{\neg\varphi^{\mathcal{R}}}, \mathcal{M}'_{\neg\varphi^{\mathcal{R}}} \subseteq \mathscr{L}$ of cvo lexicographic models for a given statement $\varphi^{\mathcal{R}}$ that help determine in which cases $\neg\varphi^{\mathcal{R}}$ is strongly compositional and when a model ($\models^*$-) satisfies $\neg\varphi^{\mathcal{R}}$.

Define $\mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ to be the set of models $\pi \in \mathscr{L}$ such that either (i) $\pi \not\models \varphi^{\mathcal{R}}$ or (ii) for every variable $X \in V_\pi$, if $\mathcal{R}^{\downarrow X}$ is acyclic then $\mathcal{R}^{\downarrow X} \subseteq =$.

Define $\mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$ to be the set of models $\pi \in \mathscr{L}$ such that either (i) $\pi \not\models \varphi^{\mathcal{R}}$ or (ii) there is no variable $X \in V_\pi$ such that $\mathcal{R}^{\downarrow X}$ is acyclic and irreflexive.

Note that $\mathcal{M}_{\neg\varphi^{\mathcal{R}}} \subseteq \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$.

The next two lemmas show that the composition of $\mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ models with $\neg\varphi^{\mathcal{R}}$ satisfying models is again satisfying $\neg\varphi^{\mathcal{R}}$, and $\mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$ is decreasing. Both results are needed to apply Proposition 4.25 to show the strong compositionality of a class of statements $\neg\varphi^{\mathcal{R}}$.

Lemma 4.46. *If $\pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ and $\pi' \models \neg\varphi^{\mathcal{R}}$ for $\pi' \in \mathscr{L}$, then $\pi \circ \pi' \models \neg\varphi^{\mathcal{R}}$.*

Proof. Suppose that $\pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ and $\pi' \models \neg\varphi^{\mathcal{R}}$. First consider the case when $\pi \not\models \varphi^{\mathcal{R}}$. The fact that $\varphi^{\mathcal{R}}$ is decreasing implies that $\pi \circ \pi' \not\models \varphi^{\mathcal{R}}$, i.e., $\pi \circ \pi' \models \neg\varphi^{\mathcal{R}}$. Now consider the other case, when $\pi \models \varphi^{\mathcal{R}}$. If for all $(\alpha, \beta) \in \mathcal{R}$ we have $\alpha(V_\pi) = \beta(V_\pi)$ then Lemma 4.42 implies that $\pi \circ \pi' \models \neg\varphi^{\mathcal{R}}$. Otherwise, let X be the first variable in π on which some pair in $\mathcal{R}$ differs. Then X is the first such variable in $\pi \circ \pi'$ as well. The definition of $\mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ implies that $\mathcal{R}^{\downarrow X}$ is not acyclic, and thus, $\pi \circ \pi' \models \neg\varphi^{\mathcal{R}}$, by Lemma 4.45. □

Lemma 4.47. *$\mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$ is decreasing (see Definition 4.11).*

Proof. Suppose that π' extends π, and that $\pi' \in \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$. We need to show that $\pi \in \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$. Assume that $\pi \notin \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$. Thus, $\pi \models \varphi^{\mathcal{R}}$ and there exists a variable $X \in V_\pi$ such that $\mathcal{R}^{\downarrow X}$ is acyclic and irreflexive. Since $\pi' \in \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$ and $V_{\pi'} \supseteq V_\pi$, we must have $\pi' \not\models \varphi^{\mathcal{R}}$. Because $\mathcal{R}^{\downarrow X}$ is irreflexive, for all $(\alpha, \beta) \in \mathcal{R}$, α and β differ on variable X. Thus, by Lemma 4.41, $\pi \models \varphi^{\mathcal{R}} \Leftrightarrow \pi' \models \varphi^{\mathcal{R}}$ which is a contradiction. □

Let us now define *simultaneously decisiveness* for sets $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. The next proposition shows that under this property statements $\neg\varphi^{\mathcal{R}}$ are strongly compositional.

Definition 4.19: Simultaneous Decisiveness

For $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$, we say that $\mathcal{R}$ is *simultaneously decisive* if for all $X \in \mathcal{V}$: if $\mathcal{R}^{\downarrow X}$ is acyclic then either $\mathcal{R}^{\downarrow X}$ is irreflexive or $\mathcal{R}^{\downarrow X} \subseteq =$.

Proposition 4.48. *If $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$ is simultaneously decisive then $\neg\varphi^{\mathcal{R}}$ is strongly compositional, and for all $\pi \in \mathscr{L}$, $\pi \models^* \neg\varphi^{\mathcal{R}} \iff \pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$.*

Proof. We will first show that $\mathcal{M}_{\neg\varphi^{\mathcal{R}}} = \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$. Suppose that $\pi \in \mathcal{M}'_{\neg\varphi^{\mathcal{R}}}$. If $\pi \not\models \varphi^{\mathcal{R}}$ then we clearly have $\pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$. Assume now that there is no variable $X \in V_\pi$ such that $\mathcal{R}^{\downarrow X}$ is acyclic and irreflexive. By our assumption on $\mathcal{R}$, if $\mathcal{R}^{\downarrow X}$ is acyclic then $\mathcal{R}^{\downarrow X} \subseteq =$, and thus, $\pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$.

Lemma 4.47 then implies that $\mathcal{M}_{\neg\varphi^{\mathcal{R}}}$ is decreasing, and it contains all models of $\neg\varphi^{\mathcal{R}}$. Then, Lemma 4.46 and Proposition 4.25 imply the result. □

4.3.2.3 Statements on Partial Variable Assignments

Next, we show that certain relatively expressive compact preference statements are strongly compositional. This includes forms of the statements $\varphi^{\mathcal{R}}$ from Proposition 4.39, where $\mathcal{R}$ is a set of pairs of alternatives. In many natural situations, $\mathcal{R}$ can be exponentially large; in the languages discussed here, we are able to express certain exponentially large sets $\mathcal{R}$ compactly.

More specifically, we consider preference statements in the language $\mathcal{L}_{pqT}$ as defined in Section 3.2. Recall that these statements are of the form $p \rhd q \mid T$, where $\rhd$ is either $\geq$, or $\gg$ or $>$, and P, Q and T are subsets of $\mathcal{V}$, with $(P \cup Q) \cap T = \emptyset$, and $p \in \underline{P}$ is an assignment to P, and $q \in \underline{Q}$ is an assignment to Q. The set φ^* is defined to include all alternative pairs that the statement φ entails, i.e., $(\alpha, \beta) \in \varphi^*$ if α extends p, β extends q, and α and β agree on the values of variables T.

Also, for any statement $\varphi \in \mathcal{L}_{pqT}$ equalling $p \rhd q \mid T$, the *non-strict version of* φ, $\varphi^{(\geq)}$, is defined as the statement $p \geq q \mid T$.

We can write statements $\mathcal{L}_{pqT}$ in a unique way as $ru \rhd su \mid T$, where $ru = p$ and $su = q$ and u consists of exactly the variable values that p and q agree on, i.e., $p(X) = q(X)$ if and only if $u(X) = p(X)$. Define u to be in the domain of

variables $U_\varphi \subseteq P \cap Q$, r in the domain of variables $R_\varphi = P \setminus U_\varphi$, and s in the domain of variables $S_\varphi = Q \setminus U_\varphi$.

Lemma 4.49. *Suppose $\varphi \in \mathcal{L}_{pqT}$ is such that $R_\varphi = S_\varphi$. Then φ^* is simultaneously decisive (see Definition 4.19).*

Proof. Let $\mathcal{R} = \varphi^*$ and consider any $X \in \mathcal{V}$ such that $\mathcal{R}^{\downarrow X}$ is acyclic and $\mathcal{R}^{\downarrow X} \not\subseteq =$. $X \in T \cup U_\varphi$ would imply $\mathcal{R}^{\downarrow X} \subseteq =$ and $X \in V \setminus (R_\varphi \cup S_\varphi \cup T \cup U_\varphi)$ would imply that $\mathcal{R}^{\downarrow X}$ is not acyclic. Thus, $X \in R_\varphi \cup S_\varphi = R_\varphi = S_\varphi$, and so $\mathcal{R}^{\downarrow X}$ equals $\{(r_\varphi(X), s_\varphi(X))\}$, which is irreflexive since $r_\varphi(X) \neq s_\varphi(X)$. □

Recall the definition of fully strict statements and weakly strict statements. A fully strict statement $p \gg q \mid T$ is satisfied by a preference model π if $\alpha \succ_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$. A weakly strict statement $p > q \mid T$ is satisfied if $\alpha \succcurlyeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$ and there exists $(\alpha', \beta') \in \varphi^*$ such that $\alpha' \succ_\pi \beta'$. As one would expect, both kinds of strict statements are strict versions of $\varphi^{\mathcal{R}}$. More specifically, fully strict statements correspond to ψ_1, and weakly strict statement correspond to ψ_2 in the example before Proposition 4.43.

Lemma 4.50. *Suppose that $\varphi \in \mathcal{L}_{pqT}$ is either a fully strict statement or a weakly strict statement. Then φ is a strict version of $\varphi^{\mathcal{R}}$ (see Definition 4.17), where $\mathcal{R} = \varphi^*$.*

Proof. Let $\varphi \in \mathcal{L}_{pqT}$ be a fully strict statement. If $\pi \models \varphi$ for $\pi \in \mathscr{L}$, then $\alpha \succ_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$. In particular, $\alpha \succeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$, i.e., $\pi \models \varphi^{\mathcal{R}}$. Thus, $\varphi^{\mathcal{R}}$ is a relaxation of φ. Also, if $\pi \models \varphi$ then $\succ_\pi = \mathcal{R}$. So if $\pi' \models \varphi^{\mathcal{R}}$ and $\succ_{\pi'} \supseteq (\succ_\pi \cap \mathcal{R})$ for some $\pi' \in \mathscr{L}$, then $\succ_{\pi'} \supseteq \mathcal{R}$ and thus $\pi' \models \varphi$. Hence, a fully strict statement φ is a strict version of $\varphi^{\mathcal{R}}$.

Now, let $\varphi \in \mathcal{L}_{pqT}$ be a weakly strict statement. If $\pi \models \varphi$ for $\pi \in \mathscr{L}$, then $\alpha \succeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$, i.e., $\pi \models \varphi^{\mathcal{R}}$. Thus, $\varphi^{\mathcal{R}}$ is a relaxation of φ. Also, if $\pi \models \varphi$ then $\succ_\pi \cap \mathcal{R} \neq \emptyset$. So if for $\pi' \in \mathscr{L}$ $\pi' \models \varphi^{\mathcal{R}}$, i.e., $\alpha \succeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$, and $\succ_{\pi'} \supseteq (\succ_\pi \cap \mathcal{R})$, then $\pi' \models \varphi$. Hence, a weakly strict statement φ is a strict version of $\varphi^{\mathcal{R}}$. □

Proposition 4.39 can be seen to imply that the non-strict elements of the language $\mathcal{L}_{pqT}$ are strongly compositional. In fact, this also holds for both kinds of strict statements and certain negations.

Theorem 4.2: (Strong) Compositionality of $\mathcal{L}_{pqT}$ and Negations

Consider any $\varphi \in \mathcal{L}_{pqT}$. Then φ is strongly compositional and $\pi \models^* \varphi$ if and only if $\pi \models \varphi^{(\geq)}$. If φ is non-strict, then $\neg\varphi$ is compositional. If φ is non-strict and also $R_\varphi = S_\varphi$, then $\neg\varphi$ is strongly compositional, and [$\pi \models^* \neg\varphi$ if and only if either $\pi \models \neg\varphi$ or $V_\pi \cap S_\varphi = \emptyset$].

Proof. Let $\mathcal{R} = \varphi^*$. First suppose that φ is either a fully strict statement or a weakly strict statement. For all $\pi \in \mathscr{L}$ we have $\pi \models \varphi^{(\geq)} \iff \pi \models \varphi^{\mathcal{R}}$. By Lemma 4.50, φ is a strict version of $\varphi^{\mathcal{R}}$. Proposition 4.43 implies that φ is strongly compositional and, for $\pi \in \mathscr{L}$, $\pi \models^* \varphi$ if and only if $\pi \models \varphi^{\mathcal{R}}$, which is if and only if $\pi \models \varphi^{(\geq)}$.

Now suppose that φ is non-strict. Then for all $\pi \in \mathscr{L}$ we have $\pi \models \varphi \iff \pi \models \varphi^{\mathcal{R}}$, and thus also, $\pi \models^* \varphi \iff \pi \models^* \varphi^{\mathcal{R}}$. Proposition 4.39 implies that $\varphi^{\mathcal{R}}$ is strongly compositional, and $\pi \models^* \varphi^{\mathcal{R}} \iff \pi \models \varphi^{\mathcal{R}}$ for any $\pi \in \mathscr{L}$. Thus, φ is strongly compositional, and, for all $\pi \in \mathscr{L}$ we have $\pi \models^* \varphi \iff \pi \models \varphi^{(\geq)}$, since $\varphi^{(\geq)} = \varphi$. Proposition 4.40 implies that $\neg\varphi$ is compositional.

Now, assume also that $R_\varphi = S_\varphi$. Then $\mathcal{R} = \varphi^*$ is simultaneously decisive, by Lemma 4.49. Proposition 4.48 implies that $\neg\varphi^{\mathcal{R}}$ and thus $\neg\varphi$ is strongly compositional and for all $\pi \in \mathscr{L}$, $\pi \models^* \neg\varphi \iff \pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}}$. We have: $\pi \in \mathcal{M}_{\neg\varphi^{\mathcal{R}}} \iff$ either (i) $\pi \not\models \varphi^{\mathcal{R}}$ or (ii) for every variable $X \in V_\pi$, either $\mathcal{R}^{\downarrow X}$ is not acyclic or $\mathcal{R}^{\downarrow X} \subseteq =$. (ii) holds if and only if for every $X \in V_\pi$, we have $X \notin R_\varphi \cup S_\varphi$ (i.e., $X \notin S_\varphi$, since $R_\varphi = S_\varphi$), so (ii) holds if and only if $V_\pi \cap S_\varphi = \emptyset$. Thus, $\pi \models^* \neg\varphi$ holds if and only if either $\pi \models \neg\varphi$ or $V_\pi \cap S_\varphi = \emptyset$. □

Theorem 4.2 suggests the feasibility of checking consistency of subsets of the language $\mathcal{L}'_{pqT}$, which is $\mathcal{L}_{pqT}$ with certain negated statements also included. Formally, define $\mathcal{L}'_{pqT}$ to be the union of $\mathcal{L}_{pqT}$ with $\{\neg\varphi : \varphi \in \mathcal{L}_{pqT}, \varphi$ *non-strict, and* $R_\varphi = S_\varphi\}$.

We can use the method of Section 4.2.3 to determine the consistency of a set of preference statements $\Gamma \subseteq \mathcal{L}'_{pqT}$, by incrementally extending a maximal $\models^*$-model π of Γ by one more variable, and then checking whether or not $\pi \models \Gamma$ holds; this makes use of Corollary 4.18 and Proposition 4.28. A closer description of this can be found in Chapter 8 together with a more detailed description on computational methods and complexities.

4.3.3 CVO Singleton Pareto Models

To consider inference and (strong) compositionality of unspecified preference statements based on cvo singleton Pareto models $\mathscr{P}$, we assume a set of variables $\mathcal{V}$ is given so that every alternative can be described and compared by a collection of values from the variable domains (see definition of Pareto models in Section 3.3).

Recall that for a cvo singleton Pareto model $\pi = \{(X_1, \geq_1), \ldots, (X_k, \geq_k)\}$ in $\mathscr{P}$, we say α is preferred to β, $\alpha \succcurlyeq_\pi \beta$, if $\alpha(X_i) \geq_i \beta(X_i)$ for all $i = 1, \ldots, k$, where $\alpha, \beta \in \mathcal{A}$ are the associated value vectors of two alternatives. We say α is strictly preferred to β, $\alpha \succ_\pi \beta$, if $\alpha \succcurlyeq_\pi \beta$ and there exists $i \in \{1, \ldots, k\}$ such that $\alpha(X_i) >_i \beta(X_i)$. Finally, we say α and β are incomparable, $\alpha \sim_\pi \beta$, if $\alpha \not\succcurlyeq_\pi \beta$ and $\beta \not\succcurlyeq_\pi \alpha$.

We consider the composition of two cvo singleton Pareto models $\pi = \{(X_1, \geq_1), \ldots, (X_k, \geq_k)\}$ and $\pi' = \{(X'_1, \geq'_1), \ldots, (X'_l, \geq'_l)\}$, by model π in union with the disjoint part of π' (see Lemma 4.9). Let $\{(X''_1, \geq''_1), \ldots, (X''_m, \geq''_m)\}$ be the set of the tuples in π' whose variables do not appear in π. Then the composition is given by $\pi \circ_{\mathscr{P}} \pi' = \{(X_1, \geq_1), \ldots, (X_k, \geq_k), (X''_1, \geq''_1), \ldots, (X''_m, \geq''_m)\}$. By this definition the empty model $\{\}$ is the unique minimum model of the set of cvo singleton Pareto models $\mathscr{P}$. As proven in Lemma 4.9 in the beginning of the chapter, $\circ_{\mathscr{P}}$ is a composition operator and thus all results from Sections 4.1 and 4.2 can be applied. For simplicity of notation, we abbreviate $\circ_{\mathscr{P}}$ to $\circ$ in the following.

As in the cvo lexicographic case from the previous section, a cvo singleton Pareto model π can be mapped to the set $\sigma(\pi) = V_\pi \subseteq \mathcal{V}$ of variables involved and $V_{\pi\circ\pi'} = V_\pi \cup V_{\pi'}$. If $V \subsetneq V_\pi$, we can construct a model π' that involves the variables $V_{\pi'} = V$ with exactly the same value orders as in π, such that $\pi \sqsupset \pi'$. Furthermore, if $V_{\pi'} \subseteq V_\pi$ for models π and π', then $\pi = \pi \circ \pi'$. Thus, the variable mapping and composition satisfy properties (i)-(iii) of Definition 4.14 and all results from Section 4.3.1 (Propositions 4.27 — 4.29 and Lemma 4.31) hold.

4.3.3.1 Statements on Complete Variable Assignments

In the following, we show that non-strict and strict statements are strongly compositional for models $\mathscr{P}$, while incomparability statements are only compositional.

The next proposition shows that if two alternatives are equivalent in one maximal CVO singleton Pareto model of Γ then they are equivalent in all models of Γ.

Proposition 4.51. *Assume that $\Gamma \subseteq \mathcal{L}$ is consistent and compositional. Let $\pi, \pi' \in \mathscr{P}$ be models of Γ and let π be a maximal model of Γ. If, for $\alpha, \beta \in \mathcal{A}$, we have $\alpha \equiv_\pi \beta$ then we have $\alpha \equiv_{\pi'} \beta$, and in fact $\Gamma \models \alpha \equiv \beta$.*

Proof. By Proposition 4.29, $V_{\pi'} \subseteq V_\pi$. Assume that $\alpha \equiv_\pi \beta$. Then $\alpha(V_\pi) = \beta(V_\pi)$, and so $\alpha(V_{\pi'}) = \beta(V_{\pi'})$ and $\pi' \models \alpha \equiv \beta$. Since π' is arbitrary, we have $\Gamma \models \alpha \equiv \beta$. □

To show that non-strict statements are strongly compositional, we make use of Lemma 4.24 and first show that non-strict statements are decreasing.

Lemma 4.52. *$\alpha \geq \beta$ is decreasing.*

Proof. Let $\pi, \pi' \in \mathscr{P}$ such that π' extends π. Suppose $\alpha \succcurlyeq_{\pi'} \beta$. Then $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in π'. Since π' extends π, $\pi \subseteq \pi'$. Hence, $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in π, i.e., $\alpha \succcurlyeq_\pi \beta$. Thus, $\alpha \geq \beta$ is decreasing. □

The following Lemma states useful properties for the satisfaction relation for compositions of models similar to Lemma 4.35.

Lemma 4.53. *Let $\pi, \pi' \in \mathscr{P}$. If $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succ_{\pi'} \beta$ then $\alpha \succ_{\pi \circ \pi'} \beta$ and $\alpha \succ_{\pi' \circ \pi} \beta$. If $\alpha \equiv_\pi \beta$, then $\alpha \succcurlyeq_{\pi'} \beta \iff \alpha \succcurlyeq_{\pi \circ \pi'} \beta$; and $\alpha \succ_{\pi'} \beta \iff \alpha \succ_{\pi \circ \pi'} \beta$.*

Proof. Suppose $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succ_{\pi'} \beta$. Then by definition of $\succcurlyeq_\pi$, $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in π. By definition of $\succ_{\pi'}$, the same holds for all $(X, \geq_X)$ in π' and there exists a tuple $(Y, \geq'_Y)$ in π' such that $\alpha(Y) >'_Y \beta(Y)$, i.e., $\alpha(Y) \neq \beta(Y)$. In the case that the same variable Y also appears in a tuple $(Y, \geq_Y)$ in π, $\alpha(Y) >_Y \beta(Y)$ since $\alpha(Y) \neq \beta(Y)$ and $\geq_Y$ is a total order. Thus, $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in $\pi' \circ \pi$ and $\pi \circ \pi'$, since all $(X, \geq_X)$ in $\pi' \circ \pi$ are either in π or in π'. Hence, $\alpha \succcurlyeq_{\pi' \circ \pi} \beta$ and $\alpha \succcurlyeq_{\pi \circ \pi'} \beta$. Also, $\pi' \circ \pi$ and $\pi \circ \pi'$ both include either $(Y, \geq'_Y)$ or $(Y, \geq_Y)$ and thus $\alpha \succ_{\pi' \circ \pi} \beta$ and $\alpha \succ_{\pi \circ \pi'} \beta$.

Assume now that $\alpha \equiv_\pi \beta$, so that $\alpha(X) = \beta(X)$ for all variables X appearing in tuples of π. The model $\pi' \circ \pi$ includes only tuple $(X, \geq_X)$ which are either in π or in π'. Thus, $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in π' if and only if $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in $\pi' \circ \pi$. Hence, $\alpha \succcurlyeq_{\pi'} \beta \iff \alpha \succcurlyeq_{\pi \circ \pi'} \beta$. Also, if there exists a

variable Y with $\alpha(Y) \neq \beta(Y)$, then $(Y, \geq_Y)$ is not in π and thus is a tuple of π' if and only if $(Y, \geq_Y)$ is a tuple of $\pi' \circ \pi$. Hence, $\alpha \succ_{\pi'} \beta \iff \alpha \succ_{\pi \circ \pi'} \beta$. □

Using the previous two results, we can now prove that non-strict statements are strongly compositional.

Proposition 4.54. *$\alpha \geq \beta$ is strongly compositional.*

Proof. We first show that $\alpha \geq \beta$ is compositional. Suppose that $\alpha \succcurlyeq_\pi \beta$ and $\alpha \succcurlyeq_{\pi'} \beta$. If $\alpha \succ_\pi \beta$ or $\alpha \succ_{\pi'} \beta$ then, by Lemma 4.53, we have $\alpha \succ_{\pi \circ \pi'} \beta$, and thus also $\alpha \succcurlyeq_{\pi \circ \pi'} \beta$. So, it just remains to deal with the case where $\alpha \equiv_\pi \beta$ and $\alpha \equiv_{\pi'} \beta$. Then $\alpha(V_\pi) = \beta(V_\pi)$ and $\alpha(V_{\pi'}) = \beta(V_{\pi'})$ and so, because $V_{\pi \circ \pi'} = V_\pi \cup V_{\pi'}$, we have $\alpha(V_{\pi \circ \pi'}) = \beta(V_{\pi \circ \pi'})$, proving $\alpha \equiv_{\pi \circ \pi'} \beta$ and hence $\alpha \succcurlyeq_{\pi \circ \pi'} \beta$. Thus, $\alpha \geq \beta$ is compositional. By Lemma 4.52, $\alpha \geq \beta$ is decreasing. Thus, Lemma 4.24 implies that $\alpha \geq \beta$ is strongly compositional. □

The following proposition gives the relation between ($\models^*$-) inference of strict and non-strict statements similar to Proposition 4.37.

Proposition 4.55. *For any alternatives $\alpha, \beta \in \mathcal{A}$ and for $\pi \in \mathscr{P}$, if $\alpha \neq \beta$, $\pi \models^* \alpha > \beta \iff \pi \models \alpha \geq \beta \iff \pi \models^* \alpha \geq \beta$.*

Proof. Suppose $\alpha \neq \beta$. Then there exists a variable Y such that $\alpha(Y) \neq \beta(Y)$. For $\pi \models \alpha \geq \beta$, either $\pi \models \alpha > \beta$ (which implies $\pi \models^* \alpha > \beta$), or $\pi \models \alpha = \beta$. Consider the case of $\pi \models \alpha = \beta$ and let $\geq_Y$ be such that $\alpha(Y) > \beta(Y)$. Then $\pi \circ (Y, \geq_Y) \models \alpha > \beta$ and thus $\pi \models^* \alpha > \beta$. Hence, $\pi \models \alpha \geq \beta$ implies $\pi \models^* \alpha > \beta$.

Now let $\pi \models^* \alpha > \beta$. Then there exists a model π' with $\pi \sqsubseteq \pi'$ and $\pi' \models \alpha > \beta$. Thus, there exists a tuple $(Y, \geq_Y)$ in π' such that $\alpha(Y) >_Y \beta(Y)$, and for all other tuples $(X, \geq_X)$ in π', $\alpha(X) \geq_X \beta(X)$. Since $\pi \subseteq \pi'$, $\alpha(X) \geq_X \beta(X)$ for all $(X, \geq_X)$ in π, i.e., $\pi \models \alpha \geq \beta$. We have shown that $\pi \models^* \alpha > \beta$ is equivalent to $\pi \models \alpha \geq \beta$.

$\pi \models \alpha \geq \beta$, by definition, implies $\pi \models^* \alpha \geq \beta$. Conversely, let $\pi \models^* \alpha \geq \beta$. Then there exists a model $\pi' \sqsupseteq \pi$ with $\pi' \models \alpha \geq \beta$. By Lemma 4.52, $\alpha \geq \beta$ is decreasing and thus $\pi \models \alpha \geq \beta$. □

The following proposition gives another example of strongly compositional preference statements.

Proposition 4.56. *$\alpha > \beta$ is strongly compositional.*

Proof. Lemma 4.53 together with Proposition 4.55 implies that $\alpha > \beta$ is strongly compositional. □

Next, we show that incomparability statements are compositional.

Proposition 4.57. *$\alpha \sim \beta$ is compositional.*

Proof. Let $\pi, \pi' \in \mathscr{P}$ be models of $\alpha \sim \beta$. Then since $\beta \not\succ_\pi \alpha$, there exists $(X, \geq_X) \in \pi$ with $\alpha(X) >_X \beta(X)$. Similarly, since $\alpha \not\succ_\pi \beta$, there exists $(Y, \geq_Y) \in \pi$ with $\beta(Y) >_Y \alpha(Y)$. Both, $(X, \geq_X)$ and $(Y, \geq_Y)$ are also in $\pi \circ \pi'$. Thus, $\beta \not\succ_{\pi\circ\pi'} \alpha$ and $\alpha \not\succ_{\pi\circ\pi'} \beta$. Hence, $\pi \circ \pi' \models \alpha \sim \beta$. □

The following example shows that incomparability statements $\alpha \sim \beta$ for $\alpha, \beta \in \mathcal{A}$ are not strongly compositional.

Example 4.5: $\alpha \sim \beta$ Is Not Strongly Compositional

Let $X_1, X_2 \in \mathcal{V}$ be variables such that $\alpha(X_i) \neq \beta(X_i)$ for $i = 1, 2$. We can construct two cvo singleton Pareto models π, π' that satisfy $\alpha \sim \beta$ in the following way.

Let $\pi = \{(X_1, \geq_1), (X_2, \geq_2)\}$ with value orders such that $\alpha(X_1) >_1 \beta(X_1)$ and $\alpha(X_2) <_2 \beta(X_2)$. Let $\pi' = \{(X_1, \geq'_1), (X_2, \geq'_2)\}$ with value orders such that $\alpha(X_1) <'_1 \beta(X_1)$ and $\alpha(X_2) >'_2 \beta(X_2)$. Then π is an extension of $\bar{\pi} = \{(X_1, \geq_1)\}$, i.e., $\bar{\pi} \models^* \alpha \sim \beta$. Also, $\bar{\pi} \models \alpha > \beta$. The composition $\bar{\pi} \circ \pi' = \{(X_1, \geq_1), (X_2, \geq'_2)\}$ satisfies $\bar{\pi} \circ \pi' \models \alpha > \beta$ and thus $\bar{\pi} \circ \pi' \not\models \alpha \sim \beta$. Hence, $\alpha \sim \beta$ is not strongly compositional.

Note that for every cvo singleton Pareto model $\pi \in \mathscr{P}$ and for any alternatives $\alpha, \beta \in \mathcal{A}$, either $\alpha \succ_\pi \beta$, or $\beta \succ_\pi \alpha$, or $\alpha \equiv_\pi \beta$ (i.e., $\alpha \succcurlyeq_\pi \beta$ and $\beta \succcurlyeq_\pi \alpha$) or $\alpha \sim_\pi \beta$ (i.e., $\alpha \not\succ_\pi \beta$ and $\beta \not\succ_\pi \alpha$).

We can consider negations as introduced in the previous section for strict, non-strict and incomparability preference statements. Then a cvo singleton Pareto model π satisfies a statement

- $\neg(\alpha > \beta)$, if $\alpha \sim_\pi \beta$ or $\beta \succcurlyeq_\pi \alpha$.
- $\neg(\alpha \geq \beta)$, if $\alpha \sim_\pi \beta$ or $\beta \succ_\pi \alpha$.

- $\neg(\alpha \sim \beta)$, if $\alpha \succcurlyeq_\pi \beta$ or $\beta \succcurlyeq_\pi \alpha$.

It can be shown that both $\neg(\alpha > \beta)$ and $\neg(\alpha \geq \beta)$ are compositional but not strongly compositional. Also, $\neg(\alpha \sim \beta)$ is not compositional.

The method of deciding consistency from Section 4.2 can thus only be applied to strict and non-strict statements. Here, a minimal extension, as for the cvo lexicographic case, includes exactly one more variable and the unique minimum model is the empty model $\{\}$.

4.3.3.2 Statements on Partial Variable Assignments

Instead of comparing complete vectors of value assignments for all variables, we can consider preference statements on partial assignments. Consider statements of the form $p \rhd q \mid T$ where p is an assignment of values to variables $P \subseteq \mathcal{V}$, q is an assignment of values to variables $Q \subseteq \mathcal{V}$, T is a set of variables disjoint from P and Q and $\rhd$ is an order relation. Let φ^* be the set of all pairs (α, β) where α extends p, β extends q, and α and β agree on T. Then a cvo singleton Pareto model π satisfies a statement φ that is

- fully strict, denoted $p \gg q \mid T$, if $\alpha \succ_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$.
- weakly strict, denoted $p > q \mid T$, if $\alpha \succcurlyeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$ and there exists $(\alpha', \beta') \in \varphi^*$ such that $\alpha' \succ_\pi \beta'$.
- non-strict, denoted $p \geq q \mid T$, if $\alpha \succcurlyeq_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$.
- fully incomparable, denoted $p \approx q \mid T$, if $\alpha \sim_\pi \beta$ for all $(\alpha, \beta) \in \varphi^*$.
- weakly incomparable, denoted $p \sim q \mid T$, if there exists $(\alpha, \beta) \in \varphi^*$ such that $\alpha \sim_\pi \beta$.
- incomparable, denoted $p \times q \mid T$, if there exists $(\alpha, \beta) \in \varphi^*$ such that $\alpha \prec_\pi \beta$ and $(\alpha', \beta') \in \varphi^*$ such that $\alpha' \succ_\pi \beta'$.

Fully strict, non-strict and fully incomparable statements can be expressed as conjunctions of strict, non-strict and incomparable statements on the tuples of alternatives in φ^*. By Lemma 4.14 together with Propositions 4.54 and 4.56, it follows that fully strict and non-strict statements are strongly compositional. Also, by Proposition 4.57, fully incomparable statements are compositional but not strongly compositional. The compositionality of the remaining statements is described in the following.

Proposition 4.58. *Weakly strict preference statements $p > q \mid T$ are strongly compositional.*

Proof. Consider a weakly strict statement $\varphi : p > q \mid T$ and cvo singleton Pareto models π, π' such that $\pi \models^* \varphi$ and $\pi' \models \varphi$. Thus there exists a model π'' extending π with $\pi'' \models \varphi$. In particular, $\alpha \succcurlyeq_{\pi''} \beta$ for all $(\alpha, \beta) \in \varphi^*$. Since by Lemma 4.52 non-strict statements $\alpha \geq \beta$ are decreasing, $\alpha \succcurlyeq_{\pi} \beta$ for all $(\alpha, \beta) \in \varphi^*$. Since $\pi' \models \varphi$, also $\alpha \succcurlyeq_{\pi'} \beta$ for all $(\alpha, \beta) \in \varphi^*$. By the strong compositionality of non-strict statements (Proposition 4.54), $\alpha \succcurlyeq_{\pi \circ \pi'} \beta$ for all $(\alpha, \beta) \in \varphi^*$. There exists $(\alpha', \beta') \in \varphi^*$ such that $\alpha' \succ_{\pi'} \beta'$ because $\pi' \models \varphi$. By Lemma 4.53 and because $\alpha' \succcurlyeq_{\pi} \beta'$, $\alpha' \succ_{\pi \circ \pi'} \beta'$. Thus, $\pi \circ \pi' \models \varphi$. □

Proposition 4.59. *Weakly incomparable preference statements $p \sim q \mid T$ are compositional, but not strongly compositional.*

Proof. Consider a weakly incomparable statement $\varphi : p \sim q \mid T$ and cvo singleton Pareto models π, π' such that $\pi \models \varphi$ and $\pi' \models \varphi$. Thus there exists $(\alpha, \beta) \in \varphi^*$ with $\alpha \sim_{\pi} \beta$. More specifically, there exists $(X, \geq_X), (X', \geq'_X) \in \pi$ with $\alpha(X) >_X \beta(X)$ and $\alpha(X') <'_X \beta(X')$. By definition $(X, \geq_X), (X', \geq'_X) \in \pi \circ \pi'$. Thus, $\alpha \sim_{\pi \circ \pi'} \beta$ and hence $\pi \circ \pi' \models \varphi$. We showed that weakly incomparable statements are compositional.

Any incomparability statement on complete alternatives $\alpha \sim \beta$ can be represented as the weakly incomparable statement $\alpha \sim \beta \mid \emptyset$. In Example 4.5 we demonstrated that $\alpha \sim \beta$ is not necessarily strongly compositional. Thus weakly incomparable statements are not necessarily strongly compositional neither. □

The next proposition shows that incomparability preference statements $p \times q \mid T$ are compositional. However, the example following the next proposition shows that incomparability preference statements are not strongly compositional.

Proposition 4.60. *Incomparability preference statements $p \times q \mid T$ are compositional.*

Proof. For any π satisfying φ, none of the variables X occurring in π or π' can be in $P_\varphi \cap Q_\varphi$ in case $p(X) \neq q(X)$, since otherwise there exists no $(\alpha, \beta) \in \varphi^*$ with $\alpha >_\pi \beta$ or alternatively there exists no $(\alpha, \beta) \in \varphi^*$ with $\alpha <_\pi \beta$.

Let π, π' be two models of φ. Suppose, $\pi \circ \pi' \not\models \varphi$ and w.o.l.g. assume that there exists no $(\alpha, \beta) \in \varphi^*$ with $\alpha >_{\pi\circ\pi'} \beta$. Since $\pi \models \varphi$, there exists $(\alpha, \beta) \in \varphi^*$ with $\alpha >_\pi \beta$. Consider the alternatives (α', β') defined as follows:

- $\alpha'(X) = \alpha(X)$ for all $X \notin (Q_\varphi \setminus P_\varphi) \cap (\sigma(\pi') \setminus \sigma(\pi))$, and $\alpha'(X) = q(X)$ for all $X \in (Q_\varphi \setminus P_\varphi) \cap (\sigma(\pi') \setminus \sigma(\pi))$,
- $\beta'(X) = \beta(X)$ if $\alpha(X) >_X \beta(X)$ for $(X, \geq_X) \in \pi$, or $X \in Q_\varphi \cup T$, and $\beta'(X) = \alpha'(X)$ otherwise.

Then $(\alpha', \beta') \in \varphi^*$, since α and thus α' extends p, and β and thus β' extends q, and α and β and thus α' and β' agree on T. Let $(X, \geq_X) \in \pi$, so that $\alpha(X) \geq_X \beta(X)$. For $\alpha(X) >_X \beta(X)$, $\alpha'(X) = \alpha(X)$ and $\beta'(X) = \beta(X)$, and thus $\alpha'(X) > \beta'(X)$. For $\alpha(X) =_X \beta(X)$, $\alpha'(X) = \alpha(X)$, and either $\beta'(X) = \beta(X) = \alpha(X)$ or $\beta'(X) = \alpha'(X) = \alpha(X)$, and thus $\alpha'(X) = \beta'(X)$. Now consider $(X, \geq_X) \in \pi' \setminus \pi$. By our observation from the beginning of the proof, $X \notin P_\varphi \cap Q_\varphi$ in case $p(X) \neq q(X)$. If $X \in P_\varphi \cap Q_\varphi$ and $p(X) = q(X)$, then $\alpha'(X) = \beta'(X)$, since α' extends p and β' extends q. If $X \in T$, then $\alpha'(X) = \beta'(X)$, since α' and β' agree on T. If $X \in Q_\varphi \setminus P_\varphi$, then $\alpha'(X) = \beta'(X)$, since $\alpha'(X) = q(X)$ and β' extends q. If $X \in \mathcal{V} \setminus (Q_\varphi \cup T)$, then $\alpha'(X) = \beta'(X)$ by definition of β'. Thus, $\alpha'(X) \geq_X \beta'(X)$ for all $(X, \geq_X) \in \pi \circ \pi'$. Also, there exists $(X, \geq_X) \in \pi \circ \pi'$ such that $\alpha'(X) >_X \beta'(X)$. Hence, $\alpha' >_{\pi\circ\pi'} \beta'$ which is a contradiction to our assumption that there exists no $(\alpha, \beta) \in \varphi^*$ with $\alpha >_{\pi\circ\pi'} \beta$. Similarly, we can prove that there exists $(\alpha, \beta) \in \varphi^*$ with $\alpha <_{\pi\circ\pi'} \beta$. We have thus proven that for $\pi \models \varphi$ and $\pi' \models \varphi$, $\pi \circ \pi' \models \varphi$, i.e., φ is compositional. □

The following example shows that incomparability statements $p \times q \mid T$ are not strongly compositional.

Example 4.6: $p \times q \mid T$ Is Not Strongly Compositional

As before, consider flight connections with variables $\mathcal{V}$ = {airline, class, time} with domains {KLM, LAN}, {business, economy} and {day, night}. Suppose a user states "Sometimes KLM flights strictly better than day-time flights with LAN, but sometimes it is exactly the other way around." This can be expressed by the preference statement φ given by (business) $\times$ (LAN, business, day) $\mid \emptyset$.

Let π be the model {(airline, LAN > KLM), (time, night > day)}. Then $\pi \models \varphi$ and π is an extension of the model π_1 = {(airline, LAN > KLM)}

such that $\pi_1 \models^* \varphi$. Let the cvo singleton Pareto model π_2 = {(airline, KLM > LAN), (time, day > night)} such that $\pi_2 \models \varphi$. Then the composition $\pi_1 \circ \pi_2$ is the model {(airline, LAN > KLM), (time, day > night)} and $\pi_1 \circ \pi_2 \not\models \varphi$. Thus, φ is not strongly compositional. The table below displays the order relation the described cvo singleton Pareto models imply on the set of tuples φ^*.

$(\alpha, \beta) \in \varphi^*$	π	π_2	$\pi_1 \circ \pi_2$
((LAN, business, day), (LAN, business, day))	$=_{\pi'}$	$=_{\pi}$	$=_{\pi' \circ \pi}$
((LAN, business, night), (LAN, business, day))	$>_{\pi'}$	$<_{\pi}$	$<_{\pi' \circ \pi}$
((KLM, business, day), (LAN, business, day))	$<_{\pi'}$	$>_{\pi}$	$<_{\pi' \circ \pi}$
((KLM, business, night), (LAN, business, day))	$\sim_{\pi'}$	$\sim_{\pi}$	$<_{\pi' \circ \pi}$

A set of fully strict, weakly strict and non-strict preference statements Γ is strongly compositional for models $\mathscr{P}$ and as described before, Γ is consistent if and only if any maximal $\models^*$-model of Γ satisfies Γ. We can construct a maximal $\models^*$-model of Γ, by starting with the empty model and iteratively finding minimal extensions that are still $\models^*$-model of Γ. Here, minimal extensions are adding exactly one tuple $(X, \geq_X)$.

4.4 Discussion

In this chapter, we have concentrated on analysing consistency and deduction for (strongly) compositional preference statements.

Here, the concept of strong compositionality, which is based on a composition operator, enables us to formulate a greedy approach to determine consistency. This approach builds up a maximal $\models^*$-model of strongly compositional preference statements Γ by iteratively finding minimal extensions of the current model, starting with a minimal model. Since all maximal $\models^*$-models of strongly compositional preference statements Γ are also models of Γ if and only if Γ is consistent, we then only need to test if the resulting model satisfies Γ. This method can be efficient, given that it is efficient to find a minimal model, to compute possible minimal extensions and to perform ($\models^*$-) satisfaction tests for a given preference model.

We showed several criteria which help to show strong compositionality of preference statements. A preference statement φ is strongly compositional if:

- it is a conjunction of strongly compositional statements,
- it is decreasing and compositional,
- there exists a set $\mathcal{M}_\varphi$ of preference models that is decreasing and contains all models of φ, and for all $\pi, \pi' \in \mathcal{G}$, if $\pi \in \mathcal{M}_\varphi$ and $\pi' \models \varphi$ then $\pi \circ \pi' \models \varphi$,
- there exists a decreasing relaxation $\bar{\varphi}$ of φ such that for all $\pi, \pi' \in \mathcal{G}$, if $\pi \models \bar{\varphi}$ and $\pi' \models \varphi$ then $\pi \circ \pi' \models \varphi$.

Furthermore, we showed several examples of natural preference statements which are strongly compositional for cvo lexicographic models $\mathscr{L}$ and cvo singleton Pareto models $\mathscr{P}$ together with specific composition operators. The same preference statements can be shown to be strongly compositional for models $\mathcal{H}(1)$ and $\mathcal{P}(1)$, respectively, under similar composition operators (by simply disregarding the value orders on variable domains).

We also showed that the minimum model for $\mathscr{L}$ and $\mathscr{P}$ (and thus $\mathcal{H}(1)$ and $\mathcal{P}(1)$) under these composition operators is simply the empty model. Also, minimal extensions consist of exactly one variable more, and all maximal models include the same variables and satisfy the same preference statements. Since testing ($\models^*$-) satisfaction is also efficient for the discussed strongly compositional preference statements, the greedy method is an efficient option to test consistency in this case. Here, for cvo lexicographic models, which imply total orders on alternatives, the same method can be used to solve the Deduction Problem. Since incomparability statements are not strongly compositional, this is not true for cvo singleton Pareto models.

More details about consistency and deduction for (fvo and cvo) lexicographic models will be given in Chapter 6 and 8. More properties and alternative algorithmic approaches for consistency and deduction for (fvo) Pareto models $\mathcal{P}(1)$ and $\mathcal{P}(t)$ with $t > 1$ are discussed in Chapter 5.

Another important type of preference models, general hierarchical models $\mathcal{H}(t)$ with $t > 1$, are discussed in Chapter 7. For general hierarchical models, we can define a composition operator similar to the composition $\circ_{\mathcal{H}(1)}$ for models $\mathcal{H}(1)$. Comparative preference statements $\mathcal{L}^{\mathcal{A}}$ on alternatives are not strongly compositional in this case (see Example 7.6), but a similar greedy search for testing consistency, in which repeatedly minimal extensions are found, can be applied, see Section 7.3.2. This method is not polynomial, due to the exponential number of minimal extensions. In fact, we can show that the Consistency

Problem for some simple preference statements is coNP-complete, which implies that there exists no composition operator for which the greedy algorithm can be polynomial.

Since strong compositionality depends on the definition of a composition operator, it is open to explore other composition operators for different preference models and test (strong) compositionality for preference statements.

In conclusion, strong compositionality is a concept that captures many different and natural types of preference statements and provides a simple method to solve the Consistency Problem, which in some cases can be very efficient.

Chapter 5

Pareto Model

Pareto orders give a natural way of comparing outcomes; one outcome is better than another if it is better on all relevant variables (different criteria by which the outcomes can be evaluated). In recommender systems and multi-objective decision making frameworks as well as the other aforementioned fields of application, one might assume that the users express their preferences (direct comparisons of two outcomes) in a Pareto manner, i.e., a user very cautiously only expresses a preference of one alternative over another if it is better or equal in all criteria. Here, one tries to find a set of optimal outcomes, i.e., outcomes that are undominated w.r.t. the Pareto order. In contrast to many other model types, deduction and consistency are not mutually expressive under Pareto models, due to the fact that Pareto models imply only partial orders on the set of outcomes. This causes the need to discuss consistency and deduction separately.

In this chapter, we consider fvo singleton Pareto models $\mathcal{P}(1)$ and k-bound Pareto models $\mathcal{P}(k)$. For better readability, we drop the mention "fvo" in the following and assume that a fixed value order for every variable domain is given.

In the next section, we first describe properties for the special case of consistency and deduction based on Pareto models that don't allow tradeoffs between variables. These properties are exploited to formulate polynomial time algorithms for PDP and PCP. One can show that the statements $\mathcal{L}^{\mathcal{A}}$ are strongly compositional for fvo Pareto models, and thus a simple greedy algorithm from Section 4.2.4 can be applied to solve the Consistency Problem. However, because consistency and deduction are not mutually expressive, this algorithm cannot be applied to solve the Deduction Problem. But since Pareto models, in

contrast to lexicographic models, do not imply an order on variables, we can give simpler characterisations of consistency and deduction. In Section 5.2.1, we develop necessary and sufficient conditions for consistency and deduction for fvo singleton Pareto models $\mathcal{P}(1)$. Similar properties for the case of consistency and deduction based on the general form of t-bound Pareto models $\mathcal{P}(t)$ with $t \geq 1$ are developed in Section 5.2.2. Here, we also show that PCP and PDP based on general Pareto models are NP-complete and coNP-complete, respectively. Throughout this chapter we will consider consistency and deduction only for the preference language $\mathcal{L}^{\mathcal{A}}$.

5.1 Preliminaries

Recall from the definition of k-bound Pareto models $\mathcal{P}(k)$ from Section 3.3, see Definition 8.3, that Pareto models are defined over a set of variables $\mathcal{V}$ by which the alternatives can be described, i.e., $\mathcal{A} = \underline{\mathcal{V}}$.

Recall from Definition 3.10 that a *k-bound Pareto model* $M \in \mathcal{P}(k)$ is a set of pairwise disjoint subsets of variables. More specifically, $M = \{C_1, \ldots, C_r\}$ with $r \geq 0$ and pairwise disjoint sets $C_i \subseteq \mathcal{V}$ with $|C_i| \leq k$ for $i = 1, \ldots, r$. When considering Pareto models in $\mathcal{P}(k)$ with $k > 1$ we will assume the variables to be commensurable so that values of different variables can be combined with the operation $\oplus$. Let D be the variable's common domain with fixed value order $\geq$. $\oplus$ is an associative, commutative and monotonic operation (where strict monotonicity means $x \oplus y \geq z \oplus y$ if $x \geq z$) on the variable's domain D. Here, $e \in D$ is the neutral element such that $e \oplus x = x$ for all $x \in D$.

The order relation on the outcomes $\mathcal{A}$ that is induced by Pareto model $M = \{C_1, \ldots, C_r\} \in \mathcal{P}(k)$ is given as in definition 3.11. For $\alpha, \beta \in \mathcal{A}$:

- $\alpha \geq_M \beta$ if $\bigoplus_{c \in C_i} \alpha(c) \geq \bigoplus_{c \in C_i} \beta(c)$ for all $i = 1, \ldots, r$. *(M satisfies $\alpha \geq \beta$, written $M \vDash \alpha \geq \beta$.)*
- $\alpha >_M \beta$ if $\alpha \geq_M \beta$ and there exists $j \in \{1, \ldots, r\}$ such that $\bigoplus_{c \in C_j} \alpha(c) > \bigoplus_{c \in C_j} \beta(c)$. *($M$ satisfies $\alpha > \beta$, or M strictly satisfies $\alpha \geq \beta$, written $M \vDash \alpha > \beta$.)*
- $\alpha \equiv_M \beta$ if $\alpha \geq_M \beta$ and $\beta \geq_M \alpha$. *(M satisfies $\alpha \equiv \beta$, written $M \vDash \alpha \equiv \beta$.)*

Note that in the context of hierarchical models in some parts of Chapter 7 (see also [WGO15]) an operator $\oplus$ that combines commensurable variable values

has been defined to be only monotonic (not strictly monotonic). However, the strict monotonicity property is needed to establish some important theoretical results in Section 5.2. This excludes operators like maximum or minimum, but still allows interesting operators like addition with neutral element 0, which is natural for combining, e.g., costs, distances, etc. In the special case of strictly positive variables X with $\underline{X} = D = \mathbb{Q}^{>0}$, multiplication can also be used as an operator with neutral element 1. For computational and complexity results, we assume that $x \oplus y$ can be computed in logarithmic time for $x, y \in D$ in this chapter.

Example 5.1

Consider the choice of holiday packages α, β and γ. We rate the holiday packages by the distance from the hotel to the city center d_c, the distance to the beach d_b, the costs for the hotel c_h and the travel costs c_t. The distances are categorised into far (0), medium (1) and near (2). The costs are categorised into high (0), medium (1) and low (2). The values of the four criteria for the outcomes α, β and γ are given by the following table.

	α	β	γ
d_c	0	2	1
d_b	1	1	2
c_h	2	1	0
c_t	2	1	1

To combine variables, consider the operator $\oplus$ that is the standard addition on the natural numbers. Then $\langle \mathcal{A}, \mathcal{V}, \oplus \rangle$ is a preference structure, where $\mathcal{A} = \{\alpha, \beta, \gamma\}$ is the set of outcomes and $\mathcal{V} = \{d_c, d_b, c_h, c_t\}$ is the set of variables.

In the following, we consider preference statements $\mathcal{L}^{\mathcal{A}}$ that are strict or non-strict comparisons on outcomes. Here, $\mathcal{L}^{\mathcal{A}}_{\geq}$ is the set of *non-strict preference statements* $\alpha \geq \beta$, and $\mathcal{L}^{\mathcal{A}}_{>}$ the set of *strict preference statements* $\alpha > \beta$, for all $\alpha, \beta \in \mathcal{A}$. Thus, $\mathcal{L}^{\mathcal{A}} = \mathcal{L}^{\mathcal{A}}_{\geq} \cup \mathcal{L}^{\mathcal{A}}_{>}$. We write $\varphi \in \mathcal{L}^{\mathcal{A}}$ as $\alpha_\varphi > \beta_\varphi$, if φ is strict, and as $\alpha_\varphi \geq \beta_\varphi$, if φ is non-strict. Recall from Definition 3.4 that $\Gamma^{(\geq)}$ is the non-strict version of $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, i.e., $\Gamma^{(\geq)} = \{\alpha_\varphi \geq \beta_\varphi \mid \varphi \in \Gamma\}$.

Definition 5.1: Reversed Statements

Define $\overline{\varphi}$ for a preference statement $\varphi \in \mathcal{L}^{\mathcal{A}}$ to be the statement $\beta_\varphi > \alpha_\varphi$ if φ is the non-strict statement $\alpha_\varphi \geq \beta_\varphi$, and $\beta_\varphi \geq \alpha_\varphi$ if φ is the strict statement $\alpha_\varphi > \beta_\varphi$.

Note that in the context of Pareto models, $\overline{\varphi}$ is not the same as $\neg\varphi$. Since the order relation that is given by a Pareto model is not necessarily complete, it can occur that a model M satisfies $M \vDash \neg\varphi$, i.e., $M \nvDash \varphi$, but $M \nvDash \overline{\varphi}$.

Example 5.2: continued

Consider variables, alternatives and operator on variables as described in Example 5.1. The Pareto model $M = \{\{d_c, d_b\}, \{c_h, c_t\}\}$ describes the situation in which a user allows tradeoffs between the distance to the city center and the distance to the beach, and tradeoffs between the cost of the hotel and the travel costs. This Pareto model satisfies $\beta >_M \gamma$ since $\beta(d_c) \oplus \beta(d_b) = 2+1 = 1+2 = \gamma(d_c) \oplus \gamma(d_b)$ and $\beta(c_h) \oplus \beta(c_t) = 1+1 > 0+1 = \gamma(c_h) \oplus \gamma(c_t)$. Furthermore, the induced order relation of M leaves the pairs of outcomes α, β and α, γ incomparable, i.e., $M \nvDash \alpha \geq \beta$, $M \nvDash \beta \geq \alpha$ and $M \nvDash \alpha \geq \gamma$, $M \nvDash \gamma \geq \alpha$. Thus, $M \vDash \neg\alpha \geq \beta$, but $M \nvDash (\overline{\alpha \geq \beta})$, i.e., $M \nvDash \beta \geq \alpha$.

A user that considers Pareto model $M' = \{\{d_b, c_h\}, \{c_t\}\}$ to describe his or her preferences allows tradeoffs between the distance to the beach and the costs of the hotel. Here, the user considers the travel costs separately and disregards the distance of the hotel to the city completely. This Pareto model satisfies $\alpha >_{M'} \gamma \equiv_{M'} \beta$.

Let $\mathcal{P}_\mathcal{V}$ denote the set of all Pareto models (with any cardinality bound on sets) over the set $\mathcal{V}$ of variables, i.e., $\mathcal{P}_\mathcal{V} = \bigcup_{k \leq |\mathcal{V}|} \mathcal{P}(k)$. We will abbreviate this notation to $\mathcal{P}$, when the set of variables $\mathcal{V}$ is clear from the context. In Section 5.2, we consider properties and complexity of the problems PCP and PDP based on Pareto models $\mathcal{P}_\mathcal{V}$ in general and based on the special classes of Pareto models $\mathcal{P}_\mathcal{V}(1)$ and $\mathcal{P}^s_\mathcal{V}$. The class $\mathcal{P}_\mathcal{V}(1)$ consists of Pareto models with only singleton sets, i.e., $\mathcal{P}_\mathcal{V}(1) = \{\{C_1, \ldots, C_r\} \in \mathcal{P}_\mathcal{V} \mid |C_i| = 1 \textit{ for all } i = 1, \ldots, r\}$. The class $\mathcal{P}^s_\mathcal{V}$ consists of Pareto models that contain only a single set, i.e., $\mathcal{P}^s_\mathcal{V} = \{\{C\} \in \mathcal{P}_\mathcal{V} \mid C \subset \mathcal{V}\}$. We adjust the notation where Pareto models

in $\mathcal{P}_{\mathcal{V}}(1)$ or $\mathcal{P}^s_{\mathcal{V}}$ are considered to avoid confusion, and omit the set of variables $\mathcal{V}$ when this is clear from the context.

Example 5.3: continued

Consider variables, alternatives and operator on variables as described in Example 5.1. Let $\Gamma = \{\alpha > \beta, \alpha \geq \gamma\}$ be a set of preference statements in $\mathcal{L}^{\mathcal{A}}$. The set Γ is consistent (for all $\mathcal{P}(k)$ with $k \geq 1$ in general and in particular for $\mathcal{P}(1)$ and for $\mathcal{P}^s$) and the following Pareto models satisfy $\alpha > \beta$ and $\alpha \geq \gamma$: $\{\{c_h\}\}$, $\{\{c_t\}\}$, $\{\{c_h\},\{c_t\}\}$, $\{\{c_h, c_t\}\}$, $\{\{c_h, d_b\}\}$, $\{\{c_h, d_b\},\{c_t\}\}$, $\{\{c_t, d_b\}\}$, $\{\{c_t, d_b\},\{c_h\}\}$, $\{\{c_h, c_t, d_b\}\}$. Furthermore, $\Gamma \not\models_{\mathcal{P}(k)} \beta \geq \gamma$ for $k > 1$ and $\Gamma \not\models_{\mathcal{P}^s} \beta \geq \gamma$ since the Pareto model $\{\{c_t, d_b\}\} \in \mathcal{P}^s \subseteq \mathcal{P}(k)$ with $k > 1$ satisfies Γ but not $\beta \geq \gamma$. However, $\Gamma \models_{\mathcal{P}(1)} \beta \geq \gamma$ since the Pareto models $\{\{c_h\}\}$, $\{\{c_t\}\}$ and $\{\{c_h\},\{c_t\}\}$ in $\mathcal{P}(1)$ all satisfy Γ and satisfy $\beta \geq \gamma$.

5.2 Properties and Solutions

For many order relations like lexicographic orders, hierarchical models and weighted sums, PDP and PCP are mutually expressive, as shown in our papers [WGO15, GRW15]. Note that for these models $\overline{\varphi}$ is equivalent to $\neg\varphi$. The following example shows that $\Gamma \cup \{\overline{\varphi}\}$ is $\mathcal{P}$-inconsistent does in general not imply $\Gamma \models_{\mathcal{P}} \varphi$ for Pareto models $\mathcal{P}$. Thus, we need to find algorithms to solve the Consistency Problem (PCP) and the Deduction Problem (PDP) separately.

Example 5.4

Let the operator $\oplus$ be the standard addition on $\mathbb{Q}^{\geq 0}$.Consider the table of values for variables c_1, c_2, c_3 evaluated at outcomes α, β, γ.

	α	β	γ
c_1	5	3	1
c_2	0	1	3
c_3	1	3	4

Let the set of given preference statements be $\Gamma = \{\gamma > \beta\}$ and let φ be the strict statement $\alpha < \beta$, so that $\overline{\varphi}$ is $\alpha \geq \beta$. The following Pareto models satisfy Γ: $\{\{c_2\}\}$, $\{\{c_3\}\}$, $\{\{c_2\},\{c_3\}\}$, $\{\{c_2, c_3\}\}$, $\{\{c_1, c_2\},\{c_3\}\}$,

$\{\{c_1, c_2, c_3\}\}$. However, none of the Γ-satisfying models satisfies $\alpha \geq \beta$. Thus, the set $\Gamma \cup \{\overline{\varphi}\} = \{\alpha \geq \beta, \gamma > \beta\}$ is $\mathcal{P}$-inconsistent. Also, $\Gamma \nvDash_{\mathcal{P}} \varphi$, as the Pareto model $\{\{c_1, c_2\}, \{c_3\}\}$ satisfies Γ but not $\alpha < \beta$. For models $\mathcal{P}^s$, $\Gamma \vDash_{\mathcal{P}^s} \varphi$.

However, we can show that $\Gamma \vDash_{\mathcal{P}} \varphi$ implies $\Gamma \cup \{\overline{\varphi}\}$ is $\mathcal{P}$-inconsistent.

Proposition 5.1. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ and $\varphi \in \mathcal{L}^{\mathcal{A}} \setminus \Gamma$ be preference statements. If $\Gamma \vDash_{\mathcal{P}} \varphi$, then $\Gamma \cup \{\overline{\varphi}\}$ is $\mathcal{P}$-inconsistent.*

Proof. Suppose $\Gamma \cup \{\overline{\varphi}\}$ is $\mathcal{P}$-consistent, i.e., there exists a Pareto model $M = \{C_1, \ldots, C_m\}$ that satisfies Γ and $M \vDash \overline{\varphi}$. Suppose φ is the strict statement $\alpha > \beta$, i.e., $\overline{\varphi}$ is the statement $\alpha \leq \beta$. Since $M \vDash \overline{\varphi}$, for all $i = 1, \ldots, m$, $\bigoplus_{c \in C_i} \alpha(c) \leq \bigoplus_{c \in C_i} \beta(c)$. Thus, $M \nvDash \varphi$, and $\Gamma \nvDash_{\mathcal{P}} \varphi$. Analogously, we can show $\Gamma \nvDash_{\mathcal{P}} \varphi$ for non-strict φ. □

5.2.1 Singleton Models

In this section, we find a simple characterisation of the Pareto inference restricted to the class $\mathcal{P}(1)$ by using set relations on sets of variables. We define the set $C_{\alpha \geq \beta} = \{c \in \mathcal{V} \mid \alpha(c) \geq \beta(c)\}$ of variables that satisfy $\alpha \geq \beta$. Similarly, $C_{\alpha > \beta} = \{c \in \mathcal{V} \mid \alpha(c) > \beta(c)\}$ and $C_{\alpha = \beta} = \{c \in \mathcal{V} \mid \alpha(c) = \beta(c)\}$. For better readability, we abbreviate the notation of a model of singleton sets $M = \{\{c_1\}, \ldots, \{c_r\}\}$ in $\mathcal{P}_{\mathcal{V}}(1)$ to $\{c_1, \ldots, c_r\}$ if the context is clear.

Note that the empty Pareto model $\{\}$ always satisfies non-strict statements, and thus, a set $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}_{\geq}$ is always $\mathcal{P}(1)$-consistent. We can prove the following characterisation of $\mathcal{P}(1)$-consistency.

Theorem 5.1: $\mathcal{P}(1)$-Consistency

Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be a set of preference statements that includes at least one strict statement. Γ is $\mathcal{P}(1)$-consistent if and only if for all strict statements $\varphi' \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$ there exists a variable c that satisfies the non-strict statements $\Gamma^{(\geq)}$ and strictly satisfies φ', i.e., $C_{\varphi'} \cap (\bigcap_{\varphi \in \Gamma^{(\geq)}} C_{\varphi}) \neq \emptyset$.

Proof. Suppose, Γ is $\mathcal{P}(1)$-consistent and let $M = \{c_1, \ldots, c_k\}$ be a Γ-satisfying model in $\mathcal{P}(1)$. Since M satisfies every statement $\varphi \in \Gamma$, $\alpha_{\varphi}(c) \geq \beta_{\varphi}(c)$ for every

$c \in M$, i.e., $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$. Furthermore, for every strict statement $\varphi' \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$ there exists a $c \in M$ such that $\alpha_{\varphi'}(c) > \beta_{\varphi'}(c)$, i.e., $c \in C_{\varphi'} \cap (\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi) \neq \emptyset$.

Conversely, suppose $C_{\varphi'} \cap (\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi) \neq \emptyset$ for all $\varphi' \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$. Consider the set $M = (\bigcup_{\varphi' \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}} C_{\varphi'}) \cap (\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi)$. For every variable $c \in M$ and every statement $\varphi \in \Gamma$, $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$, i.e., $\alpha_\varphi(c) \geq \beta_\varphi(c)$. Furthermore, for every strict statement $\varphi' \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$ there exists a $c \in M$ such that $c \in C_{\varphi'} \cap \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$, i.e., $\alpha_{\varphi'}(c) > \beta_{\varphi'}(c)$. Thus M is a Pareto model in $\mathcal{P}(1)$ that satisfies Γ, i.e., Γ is $\mathcal{P}(1)$-consistent. □

Following Theorem 5.1, we formulate the algorithm Singleton-Pareto-Consistency that solves $\mathcal{P}(1)$-PCP in polynomial time $O(|\Gamma||\mathcal{V}|)$.

Algorithm 5.1: Singleton-Pareto-Consistency(Γ,$\mathcal{V}$)

```
Input:    Variables V, statements Γ ⊆ L^A.
Question: Is Γ P(1)-consistent?
1. Let G = Γ ∩ L^A_>.
2. FOR ALL c ∈ V DO
3.     IF ( α_φ(c) ≥ β_φ(c) for all φ ∈ Γ ) THEN
4.         G = G \ {φ ∈ Γ | α_φ(c) > β_φ(c)}.
5.         IF ( G = ∅ ) THEN
6.             RETURN "Γ is consistent" and STOP.
7. RETURN "Γ is inconsistent" and STOP.
```

Proposition 5.2. *Algorithm 5.1 is correct and solves $\mathcal{P}(1)$-consistency in $O(|\Gamma||\mathcal{V}|)$.*

Proof. The correctness of Algorithm 5.1 is a direct consequence of Theorem 5.1. The for-loop is accessed $|\mathcal{V}|$ many times. Here, for every $c \in \mathcal{V}$ we test $\alpha_\varphi(c) \geq \beta_\varphi(c)$ for all $\varphi \in \Gamma$. Thus, Algorithm 5.1 runs in $O(|\Gamma||\mathcal{V}|)$. □

We can prove criteria for strict and non-strict Pareto inferences based on $\mathcal{P}(1)$ models by utilising the following lemma.

Lemma 5.3. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be a set of $\mathcal{P}(1)$-consistent preference statements. For every variable $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$ there exists a Γ-satisfying Pareto model in $\mathcal{P}(1)$ that contains c. Furthermore, for every Γ-satisfying Pareto model M in $\mathcal{P}(1)$, $M \subseteq \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$.*

Proof. Let M be a Γ-satisfying Pareto model in $\mathcal{P}(1)$ that does not contain some $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$. Since $\alpha_\varphi(c) \geq \beta_\varphi(c)$ for all $\varphi \in \Gamma$, $M \cup \{c\}$ is a Γ-satisfying Pareto model in $\mathcal{P}(1)$. Thus, for every variable $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$ there exists a Γ-satisfying Pareto model in $\mathcal{P}(1)$ that contains c. Furthermore, for every variable c in M and every $\varphi \in \Gamma$, $\alpha_\varphi(c) \geq \beta_\varphi(c)$, i.e., $c \in C_{\alpha_\varphi \geq \beta_\varphi}$. Thus, $M \subseteq \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$ for every Γ-satisfying Pareto model M in $\mathcal{P}(1)$. □

Theorem 5.2: $\mathcal{P}_\mathcal{V}(1)$-Deduction

Let $\Gamma \subseteq \mathcal{L}^\mathcal{A}$ be a set of $\mathcal{P}_\mathcal{V}(1)$-consistent preference statements. We can deduce a preference statement $\alpha \geq \beta$ from Γ ($\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha \geq \beta$) if and only if all variables $c \in \mathcal{V}$ that satisfy $\Gamma^{(\geq)}$ also satisfy $\alpha(c) \geq \beta(c)$, i.e., $\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi \subseteq C_{\alpha \geq \beta}$.

Also, Γ is $\mathcal{P}_{C_{\alpha=\beta}}(1)$-inconsistent for the set $\mathcal{P}_{C_{\alpha=\beta}}(1)$ of $\mathcal{P}(1)$ models on variables $C_{\alpha=\beta}$, if no Γ-satisfying model on variables $\mathcal{V}$ satisfies $\alpha \equiv \beta$. Then, $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha > \beta$ if and only if $\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi \subseteq C_{\alpha \geq \beta}$ and Γ is $\mathcal{P}_{C_{\alpha=\beta}}(1)$-inconsistent.

Proof. Consider the case of non-strict inference $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha \geq \beta$. By definition, $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha \geq \beta$ if and only if for every variable c involved in a Γ-satisfying Pareto model in $\mathcal{P}_\mathcal{V}(1)$, $\alpha(c) \geq \beta(c)$. By Lemma 5.3, the set of variables involved in a Γ-satisfying Pareto model in $\mathcal{P}_\mathcal{V}(1)$ is $\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$. Thus, $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha \geq \beta$ is equivalent to $c \in C_{\alpha \geq \beta}$ for all $c \in \bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi$, i.e., $\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi \subseteq C_{\alpha \geq \beta}$.

Now, consider the case of strict inference $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha > \beta$. By definition, $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha > \beta$ if and only if for every variable c involved in a Γ-satisfying Pareto model in $\mathcal{P}_\mathcal{V}(1)$, $\alpha(c) \geq \beta(c)$, and there exists no Γ-satisfying Pareto model M such that $M \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha \equiv \beta$. Thus, $\Gamma \vDash_{\mathcal{P}_\mathcal{V}(1)} \alpha > \beta$ is equivalent to $\bigcap_{\varphi \in \Gamma^{(\geq)}} C_\varphi \subseteq C_{\alpha \geq \beta}$ and there exists no Γ-satisfying Pareto model $M \in \mathcal{P}_\mathcal{V}(1)$ with $M \subseteq C_{\alpha=\beta}$, i.e., Γ is $\mathcal{P}_{C_{\alpha=\beta}}(1)$-inconsistent for the set $\mathcal{P}_{C_{\alpha=\beta}}(1)$ of $\mathcal{P}(1)$ models on variables $C_{\alpha=\beta}$. □

Following Theorem 5.2 and using the algorithm Singleton-Pareto-Consistency, we formulate the algorithm Singleton-Pareto-Deduction that solves $\mathcal{P}_\mathcal{V}(1)$-PDP in polynomial time $O(|\Gamma||\mathcal{V}|)$. Note that, for $\mathcal{P}_\mathcal{V}(1)$-inconsistent $\Gamma \subseteq \mathcal{L}^\mathcal{A}$, we can deduce any statement $\varphi \in \mathcal{L}^\mathcal{A}$.

Algorithm 5.2: Singleton-Pareto-Deduction(Γ,$\mathcal{V}$,φ)

```
Input:    Variables 𝒱, statements Γ ⊆ ℒ^𝒜 and φ ∈ ℒ^𝒜 \ Γ.
Question: Does Γ ⊨_{𝒫(1)} φ hold?
 1. IF ( Singleton-Pareto-Consistency(Γ,𝒱)
         = Γ is inconsistent ) THEN
 2.     RETURN "Γ ⊨_{𝒫(1)} φ" and STOP.
 3. Let N = ∅.
 4. FOR ALL c ∈ 𝒱 such that α_ρ(c) ≥ β_ρ(c) for all ρ ∈ Γ DO
 5.     IF ( α_φ(c) < β_φ(c) ) THEN
 6.         RETURN "Γ ⊭_{𝒫(1)} φ" and STOP.
 7.     ELSE IF ( α_φ(c) = β_φ(c) ) THEN
 8.             N = N ∪ {c}.
 8. IF ( φ ∈ ℒ^𝒜_≥ and Singleton-Pareto-Consistency(Γ,N)
         = Γ is consistent ) THEN
 9.     RETURN "Γ ⊭_{𝒫(1)} φ" and STOP.
10. ELSE RETURN "Γ ⊨_{𝒫(1)} φ" and STOP.
```

Proposition 5.4. *Algorithm 5.2 is correct and solves $\mathcal{P}(1)$-deduction in $O(|\Gamma||\mathcal{V}|)$.*

Proof. The correctness of Algorithm 5.2 is a direct consequence of Theorem 5.2. For every access of the for-loop, we test for every $c \in \mathcal{V}$ whether $\alpha_\rho(c) \geq \beta_\rho(c)$ for all $\rho \in \Gamma$. This is possible in $O(|\Gamma||\mathcal{V}|)$. Within the for-loop only constant many operations are executed. After the for-loop, we test consistency, which by Proposition 5.2 is also possible in $O(|\Gamma||\mathcal{V}|)$. Thus, Algorithm 5.2 runs in $O(|\Gamma||\mathcal{V}|)$. □

5.2.2 General Pareto Inference

In this section, we want to find characterisations for general Pareto inference, i.e., inference based on general Pareto models $\mathcal{P}$ by using set relations similar to those in the previous section. We define the set $\mathcal{V}_{\alpha\geq\beta} = \{B \subseteq \mathcal{V} \mid \bigoplus_{c\in B} \alpha(c) \geq \bigoplus_{c\in B} \beta(c)\}$ of sets of variables that satisfy $\alpha \geq \beta$. Similarly, $\mathcal{V}_{\alpha>\beta} = \{B \subseteq \mathcal{V} \mid \bigoplus_{c\in B} \alpha(c) > \bigoplus_{c\in B} \beta(c)\}$ and $\mathcal{V}_{\alpha=\beta} = \{B \subseteq \mathcal{V} \mid \bigoplus_{c\in B} \alpha(c) = \bigoplus_{c\in B} \beta(c)\}$.

As mentioned in the previous section before Theorem 5.1, a set $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}_{\geq}$ is always $\mathcal{P}(1)$-consistent and thus $\mathcal{P}$-consistent. We can prove the following characterisation of $\mathcal{P}$-consistency.

Proposition 5.5. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. Γ is $\mathcal{P}$-consistent if and only if $\bigcap_{\varphi\in\Gamma} \mathcal{V}_\varphi \neq \emptyset$.*

Proof. Suppose, $\bigcap_{\varphi\in\Gamma}\mathcal{V}_\varphi \neq \emptyset$. Then for any set $B \in \bigcap_{\varphi\in\Gamma}\mathcal{V}_\varphi$, $\{B\}$ is a Γ-satisfying Pareto model. Now suppose that Γ is $\mathcal{P}$-consistent, i.e., there exists a Γ-satisfying Pareto model $M = \{C_1, \dots, C_r\}$. For every set $C_i \in M$ and every $\varphi \in \Gamma$, $\bigoplus_{c\in C_i}\alpha_\varphi(c) \geq \bigoplus_{c\in C_i}\beta_\varphi(c)$, and for all $\varphi \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$ there exists $C_j \in M$ with $\bigoplus_{c\in C_j}\alpha_\varphi(c) > \bigoplus_{c\in C_j}\beta_\varphi(c)$. Let $C' = \bigcup_{i=1,\dots,r} C_i$. By strict monotonicity of $\oplus$, $\bigoplus_{c\in C'}\alpha_\varphi(c) \geq \bigoplus_{c\in C'}\beta_\varphi(c)$ for $\varphi \in \Gamma^{(\geq)}$, and $\bigoplus_{c\in C'}\alpha_\varphi(c) > \bigoplus_{c\in C'}\beta_\varphi(c)$ for all $\varphi \in \Gamma \cup \mathcal{L}^{\mathcal{A}}_{>}$. Thus $C' \in \bigcap_{\varphi\in\Gamma}\mathcal{V}_\varphi \neq \emptyset$. □

Remember from the preliminaries of this chapter that $\mathcal{P}^s = \{\{C\} \in \mathcal{P}_{\mathcal{V}} \mid C \subset \mathcal{V}\}$ contains all Pareto models that consist of only a single set. The proof of Proposition 5.5 directly implies the following equivalence.

Corollary 5.6. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. Γ is $\mathcal{P}$-consistent if and only if Γ is $\mathcal{P}^s$-consistent.*

Consider the relation of $\mathcal{P}$ and $\mathcal{P}^s$ for deduction. $\Gamma \models_{\mathcal{P}} \varphi$ implies $\Gamma \models_{\mathcal{P}^s} \varphi$ because $\mathcal{P}^s \subseteq \mathcal{P}$. However, Example 5.4 shows the contrary is not true.

In the following, we find characterisations for preference deduction for $\mathcal{P}_{\mathcal{V}}$. For a given set $B \subseteq \mathcal{V}$, define $\Gamma_{>B}$ to be the set of statements in Γ that are strictly satisfied by variables $B \subseteq \mathcal{V}$, i.e., $\Gamma_{>B} = \{\varphi \in \Gamma \mid \bigoplus_{c\in B}\alpha_\varphi(c) > \bigoplus_{c\in B}\beta_\varphi(c)\}$. Similarly, $\Gamma_{=B} = \{\varphi \in \Gamma \mid \bigoplus_{c\in B}\alpha_\varphi(c) = \bigoplus_{c\in B}\beta_\varphi(c)\}$. Recall that the non-strict version of preference statements Γ is denoted by $\Gamma^{(\geq)}$. We abbreviate the notation of $(\Gamma_{>B})^{(\geq)}$ to just $\Gamma^{(\geq)}_{>B}$, and define $\Gamma_{\leftrightarrow B} = (\Gamma \setminus \Gamma_{>B}) \cup \Gamma^{(\geq)}_{>B}$. Thus, $\Gamma_{\leftrightarrow B}$ replaces the preference statements in Γ that are strictly satisfied by B with their non-strict versions.

The following two propositions give characterisations for deductions. Both propositions can be proven by technical constructions. The next proposition gives a characterisation for deduction of non-strict statements.

Proposition 5.7. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be a $\mathcal{P}$-consistent set of preference statements and let $\alpha \geq \beta \in \mathcal{L}^{\mathcal{A}} \setminus \Gamma$ be a non-strict statement. $\Gamma \not\models_{\mathcal{P}_{\mathcal{V}}} \alpha \geq \beta$ if and only if there exists a set $B \in \bigcap_{\psi\in\Gamma^{(\geq)}}\mathcal{V}_\psi \cap \mathcal{V}_{\alpha<\beta}$ such that $\Gamma_{\leftrightarrow B}$ is $\mathcal{P}_{\mathcal{V}\setminus B}$-consistent, i.e., the $(\alpha \geq \beta)$-opposing set B can be extended to a Γ-satisfying Pareto model.*

Proof. Suppose, $\Gamma \not\models_{\mathcal{P}_{\mathcal{V}}} \alpha \geq \beta$. There exists a Pareto model $M = \{C_1, \dots, C_r\}$ with $M \models_{\mathcal{P}_{\mathcal{V}}} \Gamma$ and $\bigoplus_{c\in C_j}\alpha(c) < \bigoplus_{c\in C_j}\beta(c)$ for some $C_j \in M$. Since $M \models_{\mathcal{P}_{\mathcal{V}}} \Gamma$, $\bigoplus_{c\in C}\alpha_\psi(c) \geq \bigoplus_{c\in C}\beta_\psi(c)$ for all $C \in M$ and $\psi \in \Gamma$. Thus, $C_j \in \bigcap_{\psi\in\Gamma^{(\geq)}}\mathcal{V}_\psi \cap \mathcal{V}_{\alpha<\beta}$. Furthermore, for all $\gamma \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$ there exists a $C \in M$ with $\bigoplus_{c\in C}\alpha_\gamma(c) > \bigoplus_{c\in C}\beta_\gamma(c)$. In particular, for all strict statements $\gamma \in (\Gamma \setminus \Gamma_{>C_j}) \cap \mathcal{L}^{\mathcal{A}}_{>}$ (not strictly

satisfied by C_j) there exists a $C \in M \setminus \{C_j\}$ with $\bigoplus_{c \in C} \alpha_\gamma(c) > \bigoplus_{c \in C} \beta_\gamma(c)$. Thus, $M \setminus \{C_j\}$ satisfies $(\Gamma \setminus \Gamma_{>C_j}) \cup \Gamma^{(\geq)}_{>C_j}$, so that $\Gamma_{\leftrightarrow C_j}$ is $\mathcal{P}_{\mathcal{V} \setminus C_j}$-consistent.

Now suppose, there exists a set $B \in \bigcap_{\psi \in \Gamma^{(\geq)}} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha<\beta}$ such that $\Gamma_{\leftrightarrow B}$ is $\mathcal{P}_{\mathcal{V} \setminus B}$-consistent. Let $\{C_1, \ldots, C_r\}$ be a Pareto model over $\mathcal{V} \setminus B$ that satisfies $\Gamma_{\leftrightarrow B}$. Since B is in $\bigcap_{\psi \in \Gamma^{(\geq)}} \mathcal{V}_\psi$ and satisfies the statements $\Gamma_{>B}$ strictly, $\{C_1, \ldots, C_r, B\}$ is a Pareto model in $\mathcal{P}_\mathcal{V}$ that satisfies Γ. Since $B \in \mathcal{V}_{\alpha<\beta}$, the model $\{C_1, \ldots, C_r, B\}$ does not satisfy $\alpha \geq \beta$. Thus, $\Gamma \nvDash_{\mathcal{P}_\mathcal{V}} \alpha \geq \beta$. □

We prove the following characterisation for deduction of strict statements.

Proposition 5.8. *Let $\Gamma \subseteq \mathcal{L}^\mathcal{A}$ and let $\alpha > \beta \in \mathcal{L}^\mathcal{A} \setminus \Gamma$ be a strict statement. $\Gamma \nvDash_\mathcal{P} \alpha > \beta$ if and only if $\Gamma \nvDash_\mathcal{P} \alpha \geq \beta$ or $\bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta} \neq \emptyset$.*

Proof. Suppose $\Gamma \nvDash_\mathcal{P} \alpha > \beta$. Since $\alpha > \beta$ is a strict statement, this is if and only if either $\Gamma \nvDash_\mathcal{P} \alpha \geq \beta$ or there exists a Γ-satisfying Pareto model M with $M \vDash_\mathcal{P} \alpha \equiv \beta$. We show there exists a Γ-satisfying model M with $M \vDash_\mathcal{P} \alpha \equiv \beta$ if and only if $\bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta} \neq \emptyset$. Let $M = \{C_1, \ldots, C_r\} \in \mathcal{P}$ be a Γ-satisfying model with $M \vDash_\mathcal{P} \alpha \equiv \beta$. Because of strict monotonicity of $\oplus$ and because M satisfies Γ, $\bigcup_{i=1,\ldots,r} C_i \in \mathcal{V}_\psi$ for all $\psi \in \Gamma$. Furthermore, because of the monotonicity of $\oplus$, $\bigcup_{i=1,\ldots,r} C_i \in \mathcal{V}_{\alpha=\beta}$. Thus, $\bigcup_{i=1,\ldots,r} C_i \in \bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta} \neq \emptyset$.

Now suppose, $\bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta} \neq \emptyset$. Any model $M' = \{C\}$ with $C \in \bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta}$ satisfies Γ and $M' \vDash_\mathcal{P} \alpha \equiv \beta$.

Hence, $\Gamma \nvDash_\mathcal{P} \alpha > \beta$ if and only if either $\Gamma \nvDash_\mathcal{P} \alpha \geq \beta$ or $\bigcap_{\psi \in \Gamma} \mathcal{V}_\psi \cap \mathcal{V}_{\alpha=\beta} \neq \emptyset$. □

Note that the characterisation for deduction and consistency can be realised as algorithms for $\mathcal{P}$-PCP and $\mathcal{P}$-PDP, but cannot be implemented in polynomial time since this requires a search through the exponentially large power set of variables $2^\mathcal{V}$. In fact, we can prove the following complexity results for PCP and PDP.

Theorem 5.3: NP-completeness of PCP for Pareto Models

The $\mathcal{P}$-Preference Consistency Problem is NP-complete.

Proof. For any given Pareto model, we can check in polynomial time if it satisfies all given preference statements. Thus, PCP is in the class NP. We prove NP-completeness by a reduction from SAT.

Let $\mathcal{B} = K_1, \ldots, K_m$ be a set of clauses in conjunctive normal form with clauses $K_i = (l_{i,1} \vee \cdots \vee l_{i,k_i})$ for $i = 1, \ldots, m$, where the literals $l_{i,j}$ are chosen from the set of propositional variables $\mathcal{X} = \{x_1, \ldots, x_n\}$. In the following, we construct an instance of PCP from the SAT instance $\mathcal{B}$. For every propositional variable x_j, we construct three variables: p_j (corresponding to $x_j = 1$), n_j (corresponding to $x_j = 0$) and the auxiliary variable h_j. The set of variables $\mathcal{V} = \{p_j, n_j, h_j \mid j = 1, \ldots, n\}$ has cardinality polynomial in n. We define the function Q that maps the literals involved in $\mathcal{B}$ to the variables $\mathcal{V}$ by $Q(x_j) = p_j$ and $Q(\neg x_j) = n_j$. Let the set of alternatives be $\mathcal{A} = \{\alpha_i, \beta_i \mid i = 1, \ldots, m\} \cup \{\gamma_j, \delta_j, \epsilon_j, \zeta_j, \eta_j, \theta_j \mid j = 1, \ldots, n\}$. Then the cardinality of $\mathcal{A}$ is polynomial in the given sizes m and n. Let $s \in D$ with $s > e$ and $\oplus$ be an associative, commutative and strictly monotonic operation with neutral element e. Define the values of the variables on the alternatives as given in the following tables. For $i = 1, \ldots, m$ and $j = 1, \ldots, n$,

	α_i	β_i
$Q(l)$ with $l \in K_i$	s	e
$\mathcal{V} \setminus \{Q(l) \mid l \in K_i\}$	e	e

	ϵ_j	ζ_j	η_j	θ_j
p_j	s	e	e	s
n_j	s	e	e	s
h_j	e	e	s	e
$\mathcal{V} \setminus \{p_j, n_j, h_j\}$	e	e	e	e

The set $\Gamma = \{\alpha_i > \beta_i \mid i = 1, \ldots, m\} \cup \{\epsilon_j > \zeta_j, \eta_j \geq \theta_j \mid j = 1, \ldots, n\}$ of preference statements on $\mathcal{A}$ is polynomial in the given sizes m and n.

In the following, we prove that there exists a Γ-satisfying Pareto model with variables in $\mathcal{V}$ if and only if there exists a satisfying truth assignment for $\mathcal{B}$. Because of the equivalence between $\mathcal{P}$- and $\mathcal{P}^s$- consistency stated in Corollary 5.6, we can restrict the following considerations to Pareto models in $\mathcal{P}^s$.

Suppose there exists a Γ-satisfying Pareto model $M = \{C\}$ with $C \subseteq \mathcal{V}$. We prove that for each $j = 1, \ldots, n$, the set C contains either p_j or n_j and not both. Suppose for some $j \in \{1, \ldots, n\}$, $p_j \notin C$ and $n_j \notin C$. Then, $\bigoplus_{c \in C} \epsilon_j(c) = e = \bigoplus_{c \in C} \zeta_j(c)$. This contradicts $M \vDash \epsilon_j > \zeta_j$. Thus, for all $j = 1, \ldots, n$, $p_j \in C$ or $n_j \in C$. Now suppose, for some $j \in \{1, \ldots, n\}$, that $h_j \notin C$. Then, $\bigoplus_{c \in C} \eta_j(c) = e < s \leq \bigoplus_{c \in C} \theta_j(c)$. This contradicts $M \vDash \eta_j \geq \theta_j$. Thus, $h_j \in C$ for all $j = 1, \ldots, n$. Suppose, for some $j \in \{1, \ldots, n\}$, both $p_j \in C$ and $n_j \in C$. Because $h_j \in C$, $\bigoplus_{c \in C} \eta_j(c) = s < s \oplus s = \bigoplus_{c \in C} \theta_j(c)$. Again, this contradicts $M \vDash \eta_j \geq \theta_j$. Hence, for each $j = 1, \ldots, n$, M contains either $p_j \in C$ or $n_j \in C$ but not both.

Thus, for a Γ-satisfying model $M \in \mathcal{P}^s$ the assignment A, with $A(l_{i,k}) = 1$

if and only if $Q(l_{i,k}) \in M$, is well defined. Furthermore, we can show that M contains at least one variable $Q(l)$ with $l \in K_i$ for every clause with $i = 1, \ldots, m$. Suppose otherwise. Then, $\bigoplus_{c \in C} \alpha_i(c) = e \oplus \cdots \oplus e = \bigoplus_{c \in C} \beta_i(c)$. This is a contradiction to $M \vDash \alpha_i > \beta_i$. Thus, A is a satisfying truth assignment of the SAT instance $\mathcal{B}$.

Conversely, let A be a satisfying truth assignment of the Boolean formula $\mathcal{B}$. Consider the Pareto model $M = \{C\}$ with $h_j \in C$, and $p_j \in C$ if and only if $A(x_j) = 1$, and $n_j \in C$ if and only if $A(x_j) = 0$, for all $j \in \{1, \ldots, n\}$. We show $M \vDash_{\mathcal{P}} \Gamma$:

- $\alpha_i >_C \beta_i$: Since A satisfies $\mathcal{B}$, there exists $l \in \{l_{i,1}, \ldots, l_{i,k_i}\}$ for every clause K_i with $A(l) = 1$. Thus, $Q(l) \in C$ and $\bigoplus_{c \in C} \alpha_i(c) \geq s > e = \bigoplus_{c \in C} \beta_i(c)$.
- $\epsilon_j >_C \zeta_j$: Every variable x_j is assigned to be true or false. Thus either $p_j \in C$ or $n_j \in C$ (not both), and $\bigoplus_{c \in C} \epsilon_j(c) = s > e = \bigoplus_{c \in C} \delta_j(c)$.
- $\eta_j \geq_C \theta_j$: Either $p_j \in C$ or $n_j \in C$ but not both, and $h_j \in C$. Thus, $\bigoplus_{c \in C} \eta_j(c) = s = \bigoplus_{c \in C} \theta_j(c)$.

Hence, we have shown that there exists a satisfying truth assignment for $\mathcal{B}$ if and only if there exists a Γ-satisfying Pareto model in $\mathcal{P}^s_{\mathcal{V}}$, which is if and only if there exists a Γ-satisfying Pareto model in $\mathcal{P}_{\mathcal{V}}$. □

Theorem 5.4: coNP-completeness of PDP for Pareto Models

The $\mathcal{P}$-Preference Deduction Problem is coNP-complete.

Proof. For any given Pareto model, we can check in polynomial time if it satisfies all given preference statements Γ and does not satisfy φ. Thus we can verify in polynomial time that $\Gamma \nvDash \varphi$ for some instance of PDP. Hence, PDP is in the class coNP.

We prove coNP-completeness by a reduction from SAT. For a set of clauses $\mathcal{B} = K_1, \ldots, K_m$, consider the preference structure and statements as constructed in the proof of Theorem 5.3. In the following, we will define a preference statement $\varphi : \rho > \sigma$ such that no Γ-satisfying model satisfies φ. Hence, $\Gamma \vDash_{\mathcal{P}} \varphi$ if and only if Γ is $\mathcal{P}$-inconsistent, which by the previous proof is if and only if $\mathcal{B}$ is not satisfiable. For every variable $c \in \mathcal{V}$ let $\rho(c) = \sigma(c) = e$. Then every Pareto model M satisfies $M \vDash \rho = \sigma$, because every set in M evaluates to e on both ρ and σ. Thus, $M \nvDash \rho > \sigma$. □

5.3 Conclusion

We investigated the Preference Deduction Problem and the Preference Consistency Problem based on Pareto models. Here, we developed characterisations for consistency and deduction (strict and non-strict) which allow one to design algorithms for PCP and PDP. These characterisations depend on set relations of sets of supporting and opposing variables. Furthermore, we established that general $\mathcal{P}$-consistency is equivalent to $\mathcal{P}^s$-consistency, where $\mathcal{P}^s$ are models that include one single set. However, PCP and PDP are NP-complete and coNP-complete, respectively, for the general case of models $\mathcal{P}$. In the special case of singleton models, the characterisations of consistency and deduction lead to polynomial algorithms that solve PCP and PDP in $O(|\Gamma||\mathcal{V}|)$ for given preferences Γ and variables $\mathcal{V}$.

Chapter 6

FVO Lexicographic Model

In this chapter, we analyse the problems of consistency and inference based on fvo lexicographic models $\mathcal{H}(1)$ for the preference language $\mathcal{L}^{\mathcal{A}}$. For better readability, we will drop the annotation "fvo" in most places in this chapter, an always assume that fixed order relations on the variable domains are given. We will see that in this case, lexicographic models allow for efficient algorithms to solve consistency and inference. This method is a detailed description of the general algorithm formulated in the previous chapter in Section 4.2.4 for the specific case of lexicographic models and certain languages of strongly compositional statements. The general algorithm uses a greedy approach which consists of repeatedly finding minimal extensions that do not oppose any preference statement. In the following, we characterise such minimal extensions for lexicographic models $\mathcal{H}(1)$ and outline how they can be found efficiently.

Recall from Definition 3.14 that the models $\mathcal{H}(1)$ are lexicographic models over variable domains with fixed value orders, i.e., the value orders on variable domains are the same for every model. Thus, every $\pi \in \mathcal{H}(1)$ can be written as a tuple $(c_1, \ldots, c_k)$ where the set of variables involved is denoted by $\sigma(\pi) = \{c_1, \ldots, c_k\} \subseteq \mathcal{V}$. For example, we could consider variables time, class, and airline with fixed value orders day $>$ night, business $>$ economy, and KLM $>$ LAN. Then a simple lexicographic model can be fully described by giving a sequence of variables, e.g., (airline,time). The annotation of the value orders of the variables in this case can be dropped.

We start with the simplest assumption of the preference language $\mathcal{L}^{\mathcal{A}}$ that consists of statements that are strict and non-strict comparisons of alternatives, i.e., complete assignments to variables (see Definition 3.3). That is, for a set of al-

ternatives $\mathcal{A}$, $\mathcal{L}^{\mathcal{A}} = \{\alpha > \beta \mid \alpha, \beta \in \mathcal{A}\} \cup \{\alpha \geq \beta \mid \alpha, \beta \in \mathcal{A}\}$. A novelty in this chapter is the consideration of strict and non-strict preference statements together, whereas for example [Wil14] only considers non-strict preferences (however, for more general lexicographic models). Here, we also allow the set of input preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ to be an infinite set unless otherwise stated. Furthermore, we find an interesting structure, inconsistency bases, that allows us to characterise and understand inference and consistency in greater detail (Section 6.2) and solve these problems in polynomial time (Section 6.3). In Subsection 6.3.4, we briefly discuss how the outcome of the algorithm can be interpreted as the "best fitting preference model" for inconsistent preference statements. The same algorithm can be applied to decide strong consistency and max-model inference, two concepts that are discussed in Section 6.4. In Section 6.3.5, we briefly outline another preference language that enforces constraints on the importance order of variables, and show that these statements are equivalent to certain statements in the language $\mathcal{L}^{\mathcal{A}}$. We develop a proof theory and completeness results in Section 6.5 which shows the previously semantically introduced results in the perspective of logics. This is followed by a discussion at the end of the chapter.

Many parts of this chapter originate from [WGO15] and [WG17].

6.1 Preliminaries

Since the value orders of the variable domains for model $\mathcal{H}(1)$ are fixed, we will abbreviate the notation $\alpha(c) >_c \beta(c)$ to simply $\alpha(c) > \beta(c)$, where α and β are alternatives in $\mathcal{A}$ and c is a variable in $\mathcal{V}$ with associated total order $\geq_c$ on the variable domain $\underline{c}$. The Deduction Problem for lexicographic models in this case is given as follows. Given $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}}$, is it the case that $\Gamma \models_{\mathcal{H}(1)} \varphi$? That is: Is it the case that for all $\mathcal{H}(1)$-models π (over $\mathcal{A}$), if π satisfies Γ then π satisfies φ? In the case of lexicographic models with strict and non-strict preference statements in $\mathcal{L}^{\mathcal{A}}$, the Deduction Problem can be reduced to the Consistency Problem: Does there exist a model $\pi \in \mathcal{H}(1)$ such that $\pi \models_{\mathcal{H}(1)} \Gamma \cup \{\neg\varphi\}$? If the answer to this question is no, then $\Gamma \models_{\mathcal{H}(1)} \varphi$.

To show this result, we first prove that the induced order relation of $\mathcal{H}(1)$ models is complete.

Lemma 6.1. *Let $H \in \mathcal{H}(1)$ be a lexicographic model and $\alpha, \beta \in \mathcal{A}$ alternatives. Then H satisfies $\alpha \geq \beta$ if and only if H does not satisfy $\beta > \alpha$.*

Proof. Let H be the model $(c_1, \ldots, c_k)$. By definition of the lexicographic order relation, we have that H satisfies $\alpha \geq \beta$ if and only if 1) there exists $c_i \in \sigma(H)$ such that $\alpha(c_i) > \beta(c_i)$ and $\alpha(c_j) = \beta(c_j)$ for all $j < i$, or 2) $\alpha(c_j) = \beta(c_j)$ for all $j = 1, \ldots, k$. This is if and only if there exists no $c_i \in \sigma(H)$ such that $\beta(c_i) > \alpha(c_i)$ and $\beta(c_j) = \alpha(c_j)$ for all $j < i$, i.e., if and only if H does not satisfy $\beta > \alpha$. □

Proposition 6.2. *Let $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}}$. Then $\Gamma \models_{\mathcal{H}(1)} \varphi$ if and only if $\Gamma \cup \{\neg\varphi\}$ is inconsistent for models $\mathcal{H}(1)$.*

Proof. Suppose $\Gamma \models_{\mathcal{H}(1)} \varphi$. By definition of $\models_{\mathcal{H}(1)}$, this is if and only if for all models $H \in \mathcal{H}(1)$ with $H \models \Gamma$, $H \models \varphi$. Thus by Lemma 6.1, for all models $H \in \mathcal{H}(1)$ with $H \models \Gamma$, $H \not\models \neg\varphi$. This means, there does not exist any model $H \in \mathcal{H}(1)$ with $H \models \Gamma \cup \{\neg\varphi\}$, i.e., $\Gamma \cup \{\neg\varphi\}$ is inconsistent for models $\mathcal{H}(1)$. □

6.2 Inconsistency Bases

In the following we define inconsistency bases which help characterise a set of variables that cannot appear in any model of $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. First we define the support, opposition and indifference sets of variables for a preference statement $\phi \in \mathcal{L}^{\mathcal{A}}$. Here, we write $\varphi \in \mathcal{L}^{\mathcal{A}}$ as $\alpha_\varphi > \beta_\varphi$, if φ is strict, or as $\alpha_\varphi \geq \beta_\varphi$, if φ is non-strict. We consider a set $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, and a set $\mathcal{V}$ of variables by which the alternatives $\mathcal{A}$ can be specified, i.e., $\mathcal{A} = \underline{\mathcal{V}}$.

***Supp*$^\varphi_{\mathcal{V}}$, *Opp*$^\varphi$ and *Ind*$^\varphi$:** For $\varphi \in \Gamma$, define *Supp*$^\varphi$ to be $\{c \in \mathcal{V} : \alpha_\varphi(c) > \beta_\varphi(c)\}$; define *Opp*$^\varphi$ to be $\{c \in \mathcal{V} : \alpha_\varphi(c) < \beta_\varphi(c)\}$; and define *Ind*$^\varphi$ to be $\{c \in \mathcal{V} : \alpha_\varphi(c) = \beta_\varphi(c)\}$. Thus, *Supp*$^\varphi$, *Opp*$^\varphi$ and *Ind*$^\varphi$ form a partition of $\mathcal{V}$, for any $\varphi \in \mathcal{L}^{\mathcal{A}}$. Note that these three sets do not depend on whether φ is a strict statement or not. *Supp*$^\varphi$ are the variables that support φ; *Opp*$^\varphi$ are the variables that oppose φ. *Ind*$^\varphi$ are the other variables that are indifferent regarding φ.

Recall from Definition 3.15 that a model $\pi = (c_1, \ldots c_k) \in \mathcal{H}(1)$ satisfies a strict statement $\varphi \in \mathcal{L}^{\mathcal{A}}$, i.e., $\pi \models \alpha_\varphi > \beta_\varphi$, if and only if there exists some $i \in \{1, \ldots, k\}$ such that $\alpha_\varphi(c_j) = \beta_\varphi(c_j)$ for all $j < i$ and $\alpha_\varphi(c_i) > \beta_\varphi(c_i)$. π satisfies

a non-strict statement $\varphi \in \mathcal{L}^{\mathcal{A}}$, i.e., $\pi \models \alpha_\varphi \geq \beta_\varphi$, if and only if $\pi \models \alpha_\varphi > \beta_\varphi$ or $\alpha_\varphi(c_j) = \beta_\varphi(c_j)$ for all $j \in \{1, \ldots, k\}$. For a model π to satisfy φ it is necessary that no variable that opposes φ appears before all variables that support φ. More precisely, we have the following:

Lemma 6.3. *Let π be an element of $\mathcal{H}(1)$, i.e., a sequence of different elements of $\mathcal{V}$. For strict φ, $\pi \models \varphi$ if and only if an element of $Supp^\varphi$ appears in π which appears before any (if there are any) element in Opp^φ that appears. For non-strict φ, $\pi \models \varphi$ if and only if an element of $Supp^\varphi$ appears in π before any element in Opp^φ appears, or no element of Opp^φ appears in π (i.e., $\sigma(\pi) \cap Opp^\varphi = \emptyset$).*

Proof. Let $\pi = (c_1, \ldots c_k)$ be a $\mathcal{H}(1)$-model. Suppose that φ is a strict statement. Then $\pi \models \varphi$, i.e., $\alpha_\varphi \succ_\pi \beta_\varphi$, if and only if there exists some $i \in \{1, \ldots, k\}$ such that $\{c_1, \ldots c_{i-1}\} \subseteq Ind^\varphi$ and $c_i \in Supp^\varphi$, which is if and only if an element of $Supp^\varphi$ appears in π before any element in Opp^φ appears.

Now suppose that φ is a non-strict statement. Then $\pi \models \varphi$, i.e., $\alpha_\varphi \succcurlyeq_\pi \beta_\varphi$, if and only if either (i) for all $i = 1, \ldots, k$, $\alpha(c_i) = \beta(c_i)$; or (ii) there exists some $i \in \{1, \ldots, k\}$ such that $\alpha \succ_{c_i} \beta$ and for all j such that $1 \leq j < i$, $\alpha(c_j) = \beta(c_j)$. (i) holds if and only if $\sigma(\pi) \subseteq Ind^\varphi$, i.e., no element of $Supp^\varphi$ or Opp^φ appears in π. (ii) holds if and only if an element of $Supp^\varphi$ appears in π before any element in Opp^φ appears, and some element of $Supp^\varphi$ appears in π. Thus, $\pi \models \varphi$ holds if and only if either no element in Opp^φ appears in π or some element of $Supp^\varphi$ appears in π and the first such element appears before any element in Opp^φ appears. □

The following defines *inconsistency bases*, which are concerned with variables that cannot appear in any model satisfying a set of preference statements Γ (see Proposition 6.4 below). They are a valuable tool in understanding the structure of the set of satisfying models (see e.g., Proposition 6.19 below).

Definition 6.1: Inconsistency Base

Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, and let $\mathcal{V}$ be a set of variables. We say that (Γ', C') is an inconsistency base for $(\Gamma, \mathcal{V})$ if $\Gamma' \subseteq \Gamma$, and $C' \subseteq \mathcal{V}$, and

(i) for all $\varphi \in \Gamma'$, $Supp^\varphi \cup Opp^\varphi \subseteq C'$ (and thus $\mathcal{V} - C' \subseteq Ind^\varphi$); and

(ii) for all $c \in C'$, there exists $\varphi \in \Gamma'$ such that $Opp^\varphi \ni c$.

Thus, for all $\varphi \in \Gamma'$, the set C' contains all variables that are not indifferent regarding φ, and for all $c \in C'$ there is some element of Γ' that is opposed by c.

Example 6.1

Consider variables $\mathcal{V} = \{e, f, g, h\}$ with with the natural numbers as variable domains and the usual order relation on natural numbers. The values for alternatives α, β, γ and δ are given by the following table.

	α	β	γ	δ
e	1	1	0	0
f	3	0	2	2
g	3	1	1	3
h	2	2	0	1

Consider the strict preference statement $\varphi_1 : \alpha > \beta$, and the non-strict preference statements $\varphi_2 : \beta \geq \gamma$, $\varphi_3 : \gamma \geq \delta$. Let $\Gamma = \{\varphi_1, \varphi_2, \varphi_3\}$.

Then, $\mathit{Opp}^{\varphi_1} = \emptyset$, $\mathit{Supp}^{\varphi_1} = \{f, g\}$ and $\mathit{Ind}^{\varphi_1} = \{e, h\}$. Similarly, $\mathit{Opp}^{\varphi_2} = \{f\}$, $\mathit{Supp}^{\varphi_2} = \{e, h\}$ and $\mathit{Ind}^{\varphi_2} = \{g\}$. For φ_3, $\mathit{Opp}^{\varphi_3} = \{g, h\}$, $\mathit{Supp}^{\varphi_3} = \emptyset$ and $\mathit{Ind}^{\varphi_3} = \{e, f\}$.

The lexicographic model (e, f) satisfies Γ. As stated in Lemma 5, the variable $e \in \mathit{Supp}^{\varphi_2}$ precedes $f \in \mathit{Opp}^{\varphi_2}$.

Consider the tuple $(\Gamma', C') = (\{\varphi_3\}, \{g, h\})$. Condition (i) of Definition 1 is satisfied by $\mathit{Supp}^{\varphi_3} \cup \mathit{Opp}^{\varphi_3} = \{g, h\} \subseteq C'$. Since for $f, h \in C'$, $f \in \mathit{Opp}^{\varphi_3}$ and $h \in \mathit{Opp}^{\varphi_3}$, condition (ii) is satisfied as well. Thus, $(\Gamma', C') = (\{\varphi_3\}, \{g, h\})$ is an inconsistency base of $(\Gamma, \mathcal{V})$.

The following result motivates the definition of inconsistency bases (Γ', C'), showing that no model of Γ can involve any element of C', and that if Γ' contains a strict element then Γ is $\mathcal{H}(1)$-inconsistent.

Proposition 6.4. *Let (Γ', C') be an inconsistency base for $(\Gamma, \mathcal{V})$. Let π be an element of $\mathcal{H}(1)$. If $\pi \models \Gamma'$, then $C' \cap \sigma(\pi) = \emptyset$ and for any $\varphi \in \Gamma'$, $\alpha_\varphi \equiv_\pi \beta_\varphi$, so $\pi \not\models \alpha_\varphi > \beta_\varphi$. In particular, no $\mathcal{H}(1)$ model of Γ can involve any element of C'. Also, if Γ is $\mathcal{H}(1)$-consistent then Γ' contains no strict preference statements.*

Proof. Let (Γ', C') be an inconsistency base for $(\Gamma, \mathcal{V})$. Let $\pi = (c_1, \ldots, c_k)$ be an element of $\mathcal{H}(1)$ with $\pi \models \Gamma'$. Suppose π contains some element in C' and let c_i

be the element in $C' \cap \sigma(\pi)$ with the smallest index. By Definition 6.1(ii), there exists $\varphi \in \Gamma'$ such that $\mathit{Opp}^{\varphi} \ni c_i$. Furthermore, since $c_j \notin C'$ for all $1 \leq j < i$, Definition 6.1(i) implies $c_j \in \mathit{Ind}^{\varphi}$. But then, a variable that opposes φ appears before all variables that support φ. By Lemma 6.3, this is a contradiction to $\pi \models \Gamma'$; hence we must have $C' \cap \sigma(\pi) = \emptyset$. Also, for all $\varphi \in \Gamma'$, $\sigma(\pi) \subseteq \mathcal{V} \setminus C' \subseteq \mathit{Ind}^{\varphi}$ by Definition 6.1(i). Therefore, for any $\varphi \in \Gamma'$, $\alpha_\varphi \equiv_\pi \beta_\varphi$, and thus $\pi \not\models \alpha_\varphi > \beta_\varphi$. Since $\pi \models \Gamma'$, this implies that Γ' contains no strict elements. The last parts follow from the fact that Γ' is a subset of Γ, so if $\pi \models \Gamma$ then $\pi \models \Gamma'$. □

We next give a small technical lemma that will be useful later. In particular, part (i) will be used in proving compactness of preference inference.

Lemma 6.5. *Assume that (Γ', C') is an inconsistency base for $(\Gamma, \mathcal{V})$. Then the following hold.*

(i) There exists a finite set $\Gamma'' \subseteq \Gamma$ such that (Γ'', C') is an inconsistency base for $(\Gamma, \mathcal{V})$, and if Γ' contains a strict statement then Γ'' does also.

(ii) For any Δ such that $\Gamma' \subseteq \Delta \subseteq \Gamma$, (Γ', C') is an inconsistency base for $(\Delta, \mathcal{V})$.

Proof. (i): By condition (ii) of the definition of an inconsistency base, for each $c \in C'$, there exists $\varphi_c \in \Gamma'$ such that $\mathit{Opp}_{\mathcal{V}}^{\varphi_c} \ni c$. If Γ' contains a strict statement ψ, then let $\Gamma'' = \{\psi\} \cup \{\varphi_c : c \in C'\}$; else let $\Gamma'' = \{\varphi_c : c \in C'\}$. Because $\mathcal{V}$ is finite, Γ'' is finite. The definition implies that (Γ'', C') is an inconsistency base for $(\Gamma, \mathcal{V})$.

Part (ii) follows immediately from Definition 6.1, since conditions (i) and (ii) of the definition do not directly refer to Γ, but just to Γ', which is a subset of Γ. □

We will show there is, in a natural sense, a unique maximal inconsistency base for $(\Gamma, \mathcal{V})$.

For inconsistency bases (Γ_1, C_1) and (Γ_2, C_2) for $(\Gamma, \mathcal{V})$, define $(\Gamma_1, C_1) \cup (\Gamma_2, C_2)$ to be $(\Gamma_1 \cup \Gamma_2, C_1 \cup C_2)$. More generally, for inconsistency bases (Γ_i, C_i), $i \in I$, we define $\bigcup_{i \in I}(\Gamma_i, C_i)$ to be $(\bigcup_{i \in I} \Gamma_i, \bigcup_{i \in I} C_i)$, which can easily be shown to be an inconsistency base.

Lemma 6.6. *Suppose, for some (finite or infinite) non-empty index set I, and for all $i \in I$, that (Γ_i, C_i) is an inconsistency base. Then $\bigcup_{i \in I}(\Gamma_i, C_i)$ is an inconsistency base.*

Proof. For all $i \in I$, by Definition 6.1(i), for all $\varphi \in \Gamma_i$, $\mathit{Supp}^\varphi \cup \mathit{Opp}^\varphi \subseteq C_i$; thus, for all $\varphi \in \bigcup_{i\in I} \Gamma_i$, $\mathit{Supp}^\varphi \cup \mathit{Opp}^\varphi \subseteq \bigcup_{i\in I} C_i$. This proves condition (i). To prove condition (ii): for all $i \in I$, by Definition 6.1(ii), for all $c \in C_i$, there exists $\varphi \in \Gamma_i$ such that $\mathit{Opp}^\varphi \ni c$. Thus, for all $c \in \bigcup_{i\in I} C_i$, there exists $\varphi \in \bigcup_{i\in I} \Gamma_i$ such that $\mathit{Opp}^\varphi \ni c$. □

Define $\mathit{MIB}(\Gamma, \mathcal{V})$, the *maximal inconsistency base for* $(\Gamma, \mathcal{V})$, to be the union of all inconsistency bases for $(\Gamma, \mathcal{V})$, i.e., $\bigcup\{(\Gamma', C') \in \mathcal{I}\}$, where $\mathcal{I}$ is the set of inconsistency bases for $(\Gamma, \mathcal{V})$. This is well-defined, because $\mathcal{I}$ is non-empty, since it always contains the tuple $(\emptyset, \emptyset)$.

The next result states that $\mathit{MIB}(\Gamma, \mathcal{V})$ is an inconsistency base for $(\Gamma, \mathcal{V})$.

Proposition 6.7. *$\mathit{MIB}(\Gamma, \mathcal{V})$ is an inconsistency base for $(\Gamma, \mathcal{V})$, which is maximal in the following sense: If (Γ_1, C_1) is an inconsistency base for $(\Gamma, \mathcal{V})$, then $\Gamma_1 \subseteq \Gamma^\perp$ and $C_1 \subseteq C^\perp$, where $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$.*

Proof. By Lemma 6.6, the union of an arbitrary set of inconsistency bases is an inconsistency base. Consequently, $\mathit{MIB}(\Gamma, \mathcal{V})$ is an inconsistency base. Let $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$. The definition immediately implies that if (Γ_1, C_1) is an inconsistency base for $(\Gamma, \mathcal{V})$, then $\Gamma_1 \subseteq \Gamma^\perp$ and $C_1 \subseteq C^\perp$. □

By Proposition 6.4, if Γ is $\mathcal{H}(1)$-consistent then $\Gamma^\perp$ contains no strict elements, proving the next result. The converse also holds, see Lemma 6.22 below.

Proposition 6.8. *Suppose that Γ is $\mathcal{H}(1)$-consistent, i.e., there exists a $\mathcal{H}(1)$ model of Γ. Then for any inconsistency base (Γ', C') of $(\Gamma, \mathcal{V})$, $\Gamma' \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$. In particular, if $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$ then $\Gamma^\perp \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$.*

Proof. By Proposition 6.4, for every inconsistency base (Γ', C') and $\pi \in \mathcal{H}(1)$ with $\pi \models \Gamma$, we have for any $\varphi \in \Gamma'$, $\pi \not\models \alpha_\varphi > \beta_\varphi$. Thus, if Γ is $\mathcal{H}(1)$-consistent, i.e., there exists some $\pi \in \mathcal{H}(1)$ with $\pi \models \Gamma$, then Γ' contains no strict statements. □

Example 6.2

Consider variables and preference statements as in Example 6.1. The only inconsistency bases of $(\Gamma, \mathcal{V})$ are $(\emptyset, \emptyset)$ and $(\{\varphi_3\}, \{g, h\})$. Thus, $(\{\varphi_3\}, \{g, h\})$ is the maximal inconsistency base $\mathit{MIB}(\Gamma, \mathcal{V})$ and does not contain any strict statements of Γ.

6.3 Towards a Polynomial Algorithm

We can show that the statements $\mathcal{L}^{\mathcal{A}}$ are strongly compositional for models $\mathcal{H}(1)$. Thus all results from Section 4.2 hold for this case. Furthermore, we can show that $\sigma(\pi)$ for $\pi \in \mathcal{H}(1)$ gives a variable mapping (see Definition 4.14). Thus, all results from Section 4.3.1 also hold. We will, in the following, prove the strong compositionality result amongst others and find a detailed formulation of the algorithmic approach outlined in Section 4.3.1 to solve the Consistency Problem for models $\mathcal{H}(1)$ and preference statements $\mathcal{L}^{\mathcal{A}}$.

In the following, let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be a set of input preference statements, and $\mathcal{V}$ the set of variables associated with the alternatives $\mathcal{A}$.

Define $\mathit{Opp}_{\Gamma}(c)$ (usually abbreviated to $\mathit{Opp}(c)$) to be the set of statements opposed by c, i.e., $\varphi \in \Gamma$ such that $\alpha_{\varphi}(c) < \beta_{\varphi}(c)$, and define $\mathit{Supp}_{\Gamma}(c)$ (abbreviated to $\mathit{Supp}(c)$) to be the set of statements φ of Γ supported by c, (i.e., $\alpha_{\varphi}(c) > \beta_{\varphi}(c)$). For for $C' \subseteq \mathcal{V}$, we define $\mathit{Supp}_{\Gamma}(C')$ to be the statements of Γ that are supported by some element of C', i.e., $\mathit{Supp}(C') = \bigcup_{c \in C'} \mathit{Supp}(c)$. Also, for sequence of variables $(c_1, \ldots, c_k)$, we define $\mathit{Supp}(c_1, \ldots, c_k)$ to be $\bigcup_{i=1}^{k} \mathit{Supp}(c_i)$, which equals $\mathit{Supp}(\{c_1, \ldots, c_k\})$.

We thus have $\varphi \in \mathit{Supp}(c) \iff \alpha_{\varphi}(c) > \beta_{\varphi}(c) \iff c \in \mathit{Supp}^{\varphi}$; and $\varphi \in \mathit{Opp}(c) \iff \alpha_{\varphi}(c) < \beta_{\varphi}(c) \iff c \in \mathit{Opp}^{\varphi}$.

Recall that the non-strict version of preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, $\Gamma^{(\geq)}$, is defined as the set $\{\alpha_{\varphi} \geq \beta_{\varphi} : \varphi \in \Gamma\}$, i.e., Γ where the strict statements are replaced by corresponding non-strict statements. Clearly, if $\pi \models \Gamma$ then $\pi \models \Gamma^{(\geq)}$ (since $\pi \models \alpha > \beta$ implies $\pi \models \alpha \geq \beta$).

The next lemma follows immediately, since the definition of maximal inconsistency base does not depend on whether elements of Γ are strict or not.

Lemma 6.9. *For any Γ and $\mathcal{V}$, $\mathit{MIB}(\Gamma^{(\geq)}, \mathcal{V}) = \mathit{MIB}(\Gamma, \mathcal{V})$.*

In order to determine the consistency of a set of preference statements Γ, we want a method for generating a model $\pi \in \mathcal{H}(1)$ satisfying Γ. (Determining (non-)inference can be similarly performed by generating a model satisfying $\Gamma \cup \{\neg\varphi\}$, using Proposition 6.2.) By definition, $\pi \models \Gamma^{(\geq)}$ is a necessary condition for $\pi \models \Gamma$. There is a simple necessary and sufficient condition for $\pi \models \Gamma^{(\geq)}$, where $\pi = (c_1, \ldots, c_k)$, i.e., that every $\varphi \in \Gamma$ that is opposed by c_j is supported by some earlier element in the sequence (see Proposition 6.11). This condition

together with Proposition 4.28 allows one to easily incrementally grow models of $\Gamma^{(\geq)}$, until one has a maximal model of $\Gamma^{(\geq)}$. We show in the following that only maximal models of $\Gamma^{(\geq)}$ need to be considered, because if a model π of $\Gamma^{(\geq)}$ satisfies Γ, then any maximal model of $\Gamma^{(\geq)}$ extending π satisfies Γ. Note that the results about maximal inconsistency bases allow us to show a restricted version of Corollary 4.18 for the case of lexicographic models and comparative statements on complete alternatives (see Corollary 6.21 below). So to determine consistency of Γ we just need to generate any maximal model of $\Gamma^{(\geq)}$, by adding single variables that do not oppose any so far unsatisfied statements. This can be done in a straight-forward iterative way and is the basis of the algorithm.

6.3.1 Γ-allowed sequences, i.e., models of $\Gamma^{(\geq)}$

We define the notion of a Γ-allowed sequence, which turns out to be the same as a model of $\Gamma^{(\geq)}$ (see Proposition 6.11), and a $\models^*$-satisfying model of Γ, and derive important properties (Proposition 6.15), which are useful for showing the main results about maximal Γ-allowed sequences in Section 6.3.2.

For $C \subseteq \mathcal{V}$, define $\mathit{Next}_\Gamma(C)$ to be the set of all $c \in \mathcal{V} \setminus C$ such that $\mathit{Opp}(c) \subseteq \mathit{Supp}(C)$, i.e., the set of $c \in \mathcal{V} \setminus C$ that only oppose elements in Γ that are supported by elements of C. The following result gives an equivalent condition for $c \in \mathit{Next}_\Gamma(C)$.

Lemma 6.10. *Consider any $c \in \mathcal{V}$ and $C \subseteq \mathcal{V}$. Then, $c \in \mathit{Next}_\Gamma(C)$, i.e., $\mathit{Opp}(c) \subseteq \mathit{Supp}(C)$, if and only if for all $\varphi \in \Gamma \setminus \mathit{Supp}(C)$, $c \in \mathit{Supp}^\varphi \cup \mathit{Ind}^\varphi$.*

Proof. Suppose first that $\mathit{Opp}(c) \subseteq \mathit{Supp}(C)$, and consider any $\varphi \in \Gamma \setminus \mathit{Supp}(C)$. Since $\varphi \notin \mathit{Supp}(C)$, then $\varphi \notin \mathit{Opp}(c)$, and thus, $c \notin \mathit{Opp}^\varphi$. This implies that $c \in \mathit{Supp}^\varphi \cup \mathit{Ind}^\varphi$.

Conversely, suppose that for all $\varphi \in \Gamma \setminus \mathit{Supp}(C)$, $c \in \mathit{Supp}^\varphi \cup \mathit{Ind}^\varphi$. Consider any $\varphi \in \mathit{Opp}(c)$. Then $c \in \mathit{Opp}^\varphi$ and so $c \notin \mathit{Supp}^\varphi \cup \mathit{Ind}^\varphi$. Thus, $\varphi \in \mathit{Supp}(C)$. □

Definition 6.2: Γ-Allowed Sequences

Consider an arbitrary sequence $\pi = (c_1, \ldots, c_k)$ of variables in $\mathcal{V}$. Let us say that π is a *Γ-allowed sequence (of $\mathcal{V}$)* if for all $j = 1, \ldots, k$, $c_j \in \mathit{Next}(\{c_1, \ldots, c_{j-1}\})$, i.e., $\mathit{Opp}(c_j) \subseteq \mathit{Supp}(\{c_1, \ldots, c_{j-1}\})$.

Example 6.3

Consider variables and models as in Example 6.1 and preference statements $\Gamma = \{\varphi_1, \varphi_2\}$ with $\varphi_1 : \alpha > \beta$, and $\varphi_2 : \beta \geq \gamma$. Then $\pi = (h, f, e)$ is a Γ-allowed sequence since:

- $e \in \mathit{Next}(\{h, f\})$, i.e., $\mathit{Opp}(e) = \emptyset \subseteq \mathit{Supp}(\{h, f\}) = \{\varphi_1, \varphi_2\}$.
- $f \in \mathit{Next}(\{h\})$, i.e., $\mathit{Opp}(f) = \{\varphi_2\} \subseteq \mathit{Supp}(\{h\}) = \{\varphi_2\}$.
- $h \in \mathit{Next}(\emptyset)$, i.e., $\mathit{Opp}(h) = \emptyset \subseteq \mathit{Supp}(\emptyset) = \emptyset$.

π also satisfies all preference statements in Γ.

The Γ-allowed sequences turn out to be just models of $\Gamma^{(\geq)}$.

Proposition 6.11. *Consider an arbitrary sequence $\pi = (c_1, \ldots, c_k)$ of elements of $\mathcal{V}$. Then, $\pi \models \Gamma^{(\geq)}$ if and only if π is a Γ-allowed sequence.*

Proof. Suppose that $\pi \not\models \Gamma^{(\geq)}$, so there exists some $\varphi \in \Gamma$ such that $\pi \not\models \alpha_\varphi \geq \beta_\varphi$. If all elements c_j of π were indifferent to φ (i.e., $\alpha_\varphi(c_j) = \beta_\varphi(c_j)$), then we would have $\pi \models \alpha_\varphi \geq \beta_\varphi$. Thus, some element c_j in π is not indifferent to φ. Let c_i be the first such element in the sequence of π. If it were the case that $\alpha_\varphi(c_i) > \beta_\varphi(c_i)$, then we would have $\pi \models \alpha_\varphi \geq \beta_\varphi$, so we must have $\alpha_\varphi(c_i) < \beta_\varphi(c_i)$, and thus, $\varphi \in \mathit{Opp}(c_i)$. Now, $\varphi \notin \mathit{Supp}(\{c_1, \ldots, c_{i-1}\})$, since $\alpha_\varphi(c_j) = \beta_\varphi(c_j)$ for all $j < i$, and hence, $\mathit{Opp}(c_i) \not\subseteq \mathit{Supp}(\{c_1, \ldots, c_{i-1}\})$. This shows that $c_i \notin \mathit{Next}(\{c_1, \ldots, c_{i-1}\})$, and so π is not a Γ-allowed sequence.

Conversely, suppose that for some $j \in \{1, \ldots, k\}$, $c_j \notin \mathit{Next}(\{c_1, \ldots, c_{j-1}\})$, and let c_i be the first such element in the sequence. Then for all $j < i$, $c_j \in \mathit{Next}(\{c_1, \ldots, c_{j-1}\})$. Since $c_i \notin \mathit{Next}(\{c_1, \ldots, c_{i-1}\})$, there exists some $\varphi \in \Gamma \setminus \mathit{Supp}(\{c_1, \ldots, c_{i-1}\})$ such that $\varphi \in \mathit{Opp}(c_i)$, and thus, $\alpha_\varphi(c_i) < \beta_\varphi(c_i)$. Let j be minimal such that $\alpha_\varphi(c_j) \neq \beta_\varphi(c_j)$. Since $\varphi \notin \mathit{Supp}(\{c_1, \ldots, c_{i-1}\})$, we do not have $\alpha_\varphi(c_j) > \beta_\varphi(c_j)$, so we must have $\alpha_\varphi(c_j) < \beta_\varphi(c_j)$. This implies that $\pi \not\models \alpha_\varphi \geq \beta_\varphi$, where $\alpha_\varphi \geq \beta_\varphi$ is an element of $\Gamma^{(\geq)}$, and thus $\pi \not\models \Gamma^{(\geq)}$. □

In the following, it will be important to consider models extending other models. Recall from Proposition 4.6 in Section 4.1 that a composition $\pi \circ \pi'$ of two lexicographic models $\pi = (c_1, \ldots, c_k)$ and $\pi' = (c'_1, \ldots, c'_l)$ can be defined as the sequence $c_1, \ldots, c_k$ followed by all variables $\{c''_1, \ldots, c''_m\}$ that appear in π' but not in π in the same order as they appear in π'. Based on this, the extension

$\sqsupseteq$ is defined as follows: For $\pi = (c_1, \dots, c_k)$ and $\pi' = (c'_1, \dots, c'_l)$ we say that π' extends π, $\pi \sqsubseteq \pi'$, if $l \geq k$ and for all $j = 1, \dots, k$, $c'_j = c_j$. Similarly, we say that π' strictly extends π, $\pi \sqsubset \pi'$, if $l > k$ and for all $j = 1, \dots, k$, $c'_j = c_j$.

Lemma 6.12. *The mapping of models $\pi = (c_1, \dots, c_k) \in \mathcal{H}(1)$ to the set of involved variables $\sigma(\pi) = \{c_1, \dots, c_k\}$ is a variable mapping.*

Proof. We show the three properties of variable mappings for σ step by step.

(i) Since the composition $\pi \circ \pi'$ of models $\pi, \pi' \in \mathcal{H}(1)$ consists of the model π extended by all variables in π' that do not appear in π, $\pi \circ \pi'$ is mapped to the union of variables $\sigma(\pi)$ and $\sigma(\pi')$, i.e., $\sigma(\pi \circ \pi') = \sigma(\pi) \cup \sigma(\pi')$.

(ii) For model $\pi = (c_1, \dots, c_k) \in \mathcal{H}(1)$, any model $\pi' = (c_1, \dots, c_l)$ with $l < k$ and variable set $\sigma(\pi') = \{c_1, \dots, c_l\} \subsetneq \sigma(\pi)$, satisfies $\pi \sqsupset \pi'$ by definition of the extension relation.

(iii) If $\sigma(\pi') \subseteq \sigma(\pi)$ for models π and π', then there exists no variable in π' that π can be extended by. Thus, $\pi = \pi \circ \pi'$.

□

This result allows us to apply the results of Section 4.3.1 to models $\mathcal{H}(1)$.

Recall from Definition 4.3 that for a model $\pi \in \mathcal{H}(1)$ and $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, $\pi \models^* \Gamma$ if there exists a model $\pi' \in \mathcal{H}(1)$ that extends π and $\pi' \models \Gamma$.

We can show that, for consistent Γ, models of $\Gamma^{(\geq)}$ are $\models^*$-models of Γ. By Proposition 6.11, this shows that Γ-allowed sequences are also $\models^*$-models of Γ.

Proposition 6.13. *Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be consistent. Consider an arbitrary sequence $\pi = (c_1, \dots, c_k)$ of elements of $\mathcal{V}$. Then, $\pi \models \Gamma^{(\geq)}$ if and only if $\pi \models^* \Gamma$.*

Proof. First, consider any non-strict statement φ given by $\alpha \geq \beta$ in Γ. Suppose, $\pi \models \alpha \geq \beta$, then clearly $\pi \models^* \alpha \geq \beta$. Conversely, assume $\pi \models^* \alpha \geq \beta$. Then there exists a model $\pi' \in \mathcal{H}(1)$ extending π and satisfying $\pi' \models \alpha \geq \beta$. By Lemma 6.3, this is if and only if an element of $Supp^{\varphi}$ appears in π' before any element in Opp^{φ} appears, or no element of Opp^{φ} appears in π'. The same must hold for the model π, since π is an initial sequence of variables in π'. Thus, $\pi \models \alpha \geq \beta$.

Now, consider any strict statement φ given by $\alpha > \beta$ in Γ. Suppose, $\pi \models \alpha \geq \beta$. Since Γ is consistent, there exists a model π' that satisfies $\pi' \models \alpha > \beta$.

Consider the composition $\pi \circ \pi'$ of the two models, which is an extension of π. By Lemma 6.3, $\pi \models \alpha \geq \beta$ if and only if an element of *Supp*$^\varphi$ appears in π before any element in *Opp*$^\varphi$ appears, or no element of *Opp*$^\varphi$ appears in π. Also, $\pi' \models \alpha > \beta$ if and only if an element of *Supp*$^\varphi$ appears in π', and appears before any element in *Opp*$^\varphi$ appears. Since $\pi \circ \pi'$ is the model π extended by all variable in π' that do not appear in π, an element of *Supp*$^\varphi$ must appear in $\pi \circ \pi'$, and it must appear before any element in *Opp*$^\varphi$ appears. Thus, $\pi \circ \pi' \models \alpha > \beta$, and hence, $\pi \models^* \alpha > \beta$.

Then $\pi \models^* \alpha > \beta$. Conversely, assume $\pi \models^* \alpha > \beta$. Then there exists a model $\pi' \in \mathcal{H}(1)$ extending π and satisfying $\pi' \models \alpha > \beta$. By Lemma 6.3, this is if and only if an element of *Supp*$^\varphi$ appears in π', and appears before any element in *Opp*$^\varphi$ appears. Since π is an initial sequence of variables in π', an element of *Supp*$^\varphi$ appears in π before any element in *Opp*$^\varphi$ appears, or no element of *Opp*$^\varphi$ (or *Supp*$^\varphi$) appears in π. Thus, $\pi \models \alpha \geq \beta$.

This proves that $\pi \models \Gamma^{(\geq)}$ if and only if $\pi \models^* \Gamma$. □

We can now prove the strong compositionality of statements $\mathcal{L}^{\mathcal{A}}$ in connection with models $\mathcal{H}(1)$, which allows us to apply the results of Section 4.2.

Proposition 6.14. *Let $\varphi \in \mathcal{L}^{\mathcal{A}}$. Then φ is strongly compositional for models $\mathcal{H}(1)$, i.e., for $\pi, \pi' \in \mathcal{H}(1)$ with $\pi \models^* \varphi$ and $\pi' \models \varphi$, $\pi \circ \pi' \models \varphi$.*

Proof. By Proposition 6.13 we have that $\pi \models^* \varphi$ if and only if $\pi \models^* \alpha_\varphi \geq \beta_\varphi$.

Let us first consider a non-strict statement φ written as $\alpha \geq \beta$. Then $\pi \models^* \varphi$ is the same as $\pi \models \varphi$. By Lemma 6.3, we have for π and π' that an element of *Supp*$^\varphi$ appears in the model before any element in *Opp*$^\varphi$ appears, or no element of *Opp*$^\varphi$ appears in the model. Since the composition $\pi \circ \pi'$ is an extension of π, $\pi \circ \pi'$ consists of the variables in π followed by the variables in π' that are not in π. Thus, if an element of *Opp*$^\varphi$ appears in π then there exists and element of *Supp*$^\varphi$ preceding it, and the same is true in $\pi \circ \pi'$, i.e., $\pi \circ \pi' \models \varphi$. Consider the case that no element of *Opp*$^\varphi$ appears in π. Then any element c_o of *Opp*$^\varphi$ that appears in $\pi \circ \pi'$ must appear in π'. Thus, c_o is preceded by an element c_s in *Supp*$^\varphi$ in π'. By definition of the composition operator, c_o is also preceded by c_s in $\pi \circ \pi'$. Hence, $\pi \circ \pi' \models \varphi$.

Consider now a strict statement φ written as $\alpha > \beta$. Then $\pi \models^* \varphi$ is the same as $\pi \models \alpha \geq \beta$. By Lemma 6.3, we have for π that an element of *Supp*$^\varphi$ appears in π before any element in *Opp*$^\varphi$ appears, or no element of *Opp*$^\varphi$ appears in π.

Also, we have for π' that an element of *Supp*$^\varphi$ appears in the π', and appears before any element in *Opp*$^\varphi$ appears. Thus, if an element of *Opp*$^\varphi$ appears in π then there exists and element of *Supp*$^\varphi$ preceding it, and the same is true in $\pi \circ \pi'$, i.e., $\pi \circ \pi' \models \varphi$. Consider the case that no element of *Opp*$^\varphi$ appears in π. Since there exists an element c_s in *Supp*$^\varphi$ in π', which precedes any element of *Opp*$^\varphi$ in π', c_s is also in $\pi \circ \pi'$ and precedes any element of *Opp*$^\varphi$ in $\pi \circ \pi'$. Hence, $\pi \circ \pi' \models \varphi$. □

We also have the following property of Γ-allowed sequences.

Proposition 6.15. *Suppose that π is a Γ-allowed sequence. Then, for all $\varphi \in$ Supp(π), $\pi \models \alpha_\varphi > \beta_\varphi$, and for all $\varphi \in \Gamma \setminus$ Supp(π), $\alpha_\varphi \equiv_\pi \beta_\varphi$, so, in particular $\pi \not\models \alpha_\varphi > \beta_\varphi$. Thus, for $\varphi \in \Gamma$, we have $\pi \models \alpha_\varphi > \beta_\varphi$ if and only if $\varphi \in$ Supp(π). Also, $\pi \models \Gamma$ if and only if every strict element of Γ is in Supp(π).*

Proof. First, consider any $\varphi \in$ *Supp*(π). Thus there exists $c_j \in \sigma(\pi)$ such that $\alpha_\varphi(c_j) > \beta_\varphi(c_j)$, so, in particular, $\alpha_\varphi(c_j) \neq \beta_\varphi(c_j)$. Let i be minimal such that $\alpha_\varphi(c_i) \neq \beta_\varphi(c_i)$. Proposition 6.11 implies that $\pi \models \alpha_\varphi \geq \beta_\varphi$, which implies that $\alpha_\varphi(c_i) \not< \beta_\varphi(c_i)$, and thus $\alpha_\varphi(c_i) > \beta_\varphi(c_i)$, proving that $\pi \models \alpha_\varphi > \beta_\varphi$.

Now, consider $\varphi \in \Gamma \setminus$ *Supp*(π). If it were the case that there exists $c_j \in \sigma(\pi)$ such that $\alpha_\varphi(c_j) \neq \beta_\varphi(c_j)$, then the argument above implies that there exists i such that $\alpha_\varphi(c_i) > \beta_\varphi(c_i)$, and thus $\varphi \in$ *Supp*(π). Thus, for all $c_j \in \sigma(\pi)$, $\alpha_\varphi(c_j) = \beta_\varphi(c_j)$, and, hence, $\alpha_\varphi \equiv_\pi \beta_\varphi$.

For the last part, since, by Proposition 6.11, $\pi \models \Gamma^{(\geq)}$, we have: $\pi \models \Gamma$ if and only if for every strict element φ of Γ, $\pi \models \alpha_\varphi > \beta_\varphi$, i.e., $\varphi \in$ *Supp*(π). □

6.3.2 Maximal Γ-allowed sequences

As before, when talking about maximal models, with respect to some set of models $\mathcal{G}$, we mean maximality with respect to the extension relation, so a model in $\mathcal{G}$ is ($\mathcal{G}$-)maximal if there is no element of $\mathcal{G}$ that extends it.

Definition 6.3: Maximal Γ-Allowed Sequences

We say that π is a maximal Γ-allowed sequence of $\mathcal{V}$ if π is a Γ-allowed sequence of $\mathcal{V}$ and no extension of π is a Γ-allowed sequence of $\mathcal{V}$, i.e., *Next*$(\sigma(\pi)) = \emptyset$.

Lemma 6.16. *Suppose that $\pi, \pi' \in \mathcal{H}(1)$ and $\pi, \pi' \models \Gamma^{(\geq)}$, and that π' extends π. Then for all $\varphi \in \Gamma$, if $\pi \models \varphi$ then $\pi' \models \varphi$. In particular, if $\pi \models \Gamma$ then $\pi' \models \Gamma$.*

Proof. Assume that $\pi, \pi' \models \Gamma^{(\geq)}$, and π' extends π. Consider any $\varphi \in \Gamma$, and suppose that $\pi \models \varphi$. If φ is non-strict, then $\varphi \in \Gamma^{(\geq)}$ and so $\pi' \models \varphi$. If φ is strict, then by Lemma 6.3, $\pi \models \varphi$ if and only if an element of Supp^{φ} appears in π, and appears before any element in Opp^{φ} appears in π. Since π' is an extension of π and thus has π as an initial sequence of variables, we also have that an element of Supp^{φ} appears in π', and appears before any element in Opp^{φ} appears in π'. Thus, $\pi' \models \varphi$. □

We use this in proving the next result, which shows that if we are interested in finding models of Γ it is sufficient to only consider maximal Γ-allowed sequences, i.e., maximal models of $\Gamma^{(\geq)}$.

Lemma 6.17. *If π is a Γ-allowed sequence, then either π is a maximal Γ-allowed sequence or there exists a maximal Γ-allowed sequence π' that extends π. Then, for all $\varphi \in \Gamma$, if $\pi \models \varphi$ then $\pi' \models \varphi$. In particular, if $\pi \models \Gamma$ then $\pi' \models \Gamma$.*

Proof. The *extends* relation on the finite set of Γ-allowed sequences is transitive and acyclic. It follows that for any Γ-allowed sequence π there exists a maximal Γ-allowed sequence extending π. The last part follows from the previous result, Lemma 6.16 (using the equivalence stated by Proposition 6.11). □

The following key lemma shows the close relationship between maximal Γ-allowed sequences and the maximal inconsistency base.

Lemma 6.18. *Suppose that π is a maximal Γ-allowed sequence. Then $(\Gamma \setminus \mathit{Supp}(\pi), \mathcal{V} \setminus \sigma(\pi))$ equals $\mathit{MIB}(\Gamma, \mathcal{V})$.*

Proof. We first check the two conditions in the definition of an inconsistency base (see Definition 6.1). Consider any element φ of $\Gamma \setminus \mathit{Supp}(\pi)$. Proposition 6.15 implies that $\alpha_{\varphi} \equiv_{\pi} \beta_{\varphi}$, so that for all $c \in \sigma(\pi)$, $\alpha_{\varphi}(c) = \beta_{\varphi}(c)$, and so $\sigma(\pi) \subseteq \mathit{Ind}^{\varphi}$, showing that Condition (i) holds. Now, consider any variable c in $\mathcal{V} \setminus \sigma(\pi)$. By definition of a maximal Γ-allowed sequence, $\mathit{Next}(\sigma(\pi)) = \emptyset$, so $c \notin \mathit{Next}(\sigma(\pi))$. Therefore, by Lemma 6.10, there exists $\varphi \in \Gamma \setminus \mathit{Supp}(\pi)$ such that $c \notin \mathit{Supp}^{\varphi} \cup \mathit{Ind}^{\varphi}$, so $c \in \mathit{Opp}^{\varphi}$, showing that Condition (ii) of an inconsistency base holds.

Write $\mathit{MIB}(\Gamma, \mathcal{V})$ as $(\Gamma^\perp, C^\perp)$. Thus, by definition, $\Gamma \setminus \mathit{Supp}(\pi) \subseteq \Gamma^\perp$ and $\mathcal{V} \setminus \sigma(\pi) \subseteq C^\perp$. Proposition 6.11 implies that $\pi \models \Gamma^{(\geq)}$. Lemma 6.9 implies that $\mathit{MIB}(\Gamma^{(\geq)}, \mathcal{V}) = (\Gamma^\perp, C^\perp)$. Proposition 6.4 then implies that $C^\perp \cap \sigma(\pi) = \emptyset$, and so, $\mathcal{V} \setminus \sigma(\pi) \supseteq C^\perp$. Thus, $\mathcal{V} \setminus \sigma(\pi) = C^\perp$.

Consider any $\varphi \in \Gamma^\perp$. By definition of an inconsistency base, $\mathcal{V} \setminus C^\perp \subseteq \mathit{Ind}^\varphi$, i.e., $\sigma(\pi) \subseteq \mathit{Ind}^\varphi$, which implies $\alpha_\varphi \equiv_\pi \beta_\varphi$, and so, by Proposition 6.15, $\varphi \in \Gamma \setminus \mathit{Supp}(\pi)$. Thus, $\Gamma^\perp \subseteq \Gamma \setminus \mathit{Supp}(\pi)$, and hence, $\Gamma^\perp = \Gamma \setminus \mathit{Supp}(\pi)$, completing the proof that $(\Gamma \setminus \mathit{Supp}(\pi), \mathcal{V} \setminus \sigma(\pi))$ equals $(\Gamma^\perp, C^\perp)$. □

Different maximal Γ-allowed sequences satisfy the same subset of Γ and involve the same subset of $\mathcal{V}$:

Proposition 6.19. *Suppose that π is a maximal Γ-allowed sequence. Write $\mathit{MIB}(\Gamma, \mathcal{V})$ as $(\Gamma^\perp, C^\perp)$. Then $\Gamma^\perp = \Gamma \setminus \mathit{Supp}(\pi)$ and $C^\perp = \mathcal{V} \setminus \sigma(\pi)$. Thus, if π' is another maximal Γ-allowed sequence, then $\sigma(\pi') = \sigma(\pi)$ and $\mathit{Supp}(\pi') = \mathit{Supp}(\pi)$. Also, for all $\varphi \in \Gamma$, $\pi \models \varphi \iff \pi' \models \varphi$, which is if and only if φ is not a strict element of $\Gamma^\perp$. Hence, every maximal Γ-allowed sequence satisfies the same elements of Γ.*

Proof. By Lemma 6.18, $\Gamma^\perp = \Gamma \setminus \mathit{Supp}(\pi)$ and $C^\perp = \mathcal{V} \setminus \sigma(\pi)$. For any maximal Γ-allowed sequence π', $\sigma(\pi') = \mathcal{V} \setminus C^\perp = \sigma(\pi)$, and $\mathit{Supp}(\pi') = \Gamma \setminus \Gamma^\perp = \mathit{Supp}(\pi)$.

Since by Proposition 6.11 and 6.13 the maximal Γ-allowed sequences are the same as maximal $\models^*$-models of Γ, and the statements $\mathcal{L}^{\mathcal{A}}$ are strongly compositional for models $\mathcal{H}(1)$ by Proposition 6.14, the last part follows from Theorem 4.1 (i). □

No model of $\Gamma^{(\geq)}$ satisfies any element of Γ that is not satisfied by a maximal Γ-allowed sequence π.

Proposition 6.20. *Consider any maximal Γ-allowed sequence π, and any $\pi' \in \mathcal{H}(1)$ such that $\pi' \models \Gamma^{(\geq)}$. For any $\varphi \in \Gamma$, if $\pi' \models \varphi$ then $\pi \models \varphi$.*

Proof. Let $\varphi \in \Gamma$. Suppose that $\pi \not\models \varphi$, and so, by Proposition 6.15, φ is strict and $\varphi \in \Gamma \setminus \mathit{Supp}(\pi)$. Consider any model $\pi' \models \Gamma^{(\geq)}$. By Proposition 6.11, π' is a Γ-allowed sequence. By Lemma 6.17, there exists some maximal Γ-allowed sequence π'' that extends or equals π'. We have $\mathit{Supp}(\pi') \subseteq \mathit{Supp}(\pi'')$. Proposition 6.19 implies that $\mathit{Supp}(\pi) = \mathit{Supp}(\pi'')$, so $\varphi \notin \mathit{Supp}(\pi')$. Since φ is strict, $\pi' \not\models \varphi$, again using Proposition 6.15. □

The corollary below is a restriction of Corollary 4.18, to the case of $\mathcal{H}(1)$ models and preference statements in $\mathcal{L}^{\mathcal{A}}$. Following Propositions 6.11 and 6.13, we can replace the expression "$\models^*$-model of Γ" by "Γ-allowed sequence" in the statement. The corollary then shows that to test consistency, one just needs to generate a single maximal Γ-allowed sequence (i.e., maximal $\models^*$-model of Γ), which can be easily done using an iterative algorithm.

Corollary 6.21. *Let π be any maximal Γ-allowed sequence of preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. Then Γ is $\mathcal{H}(1)$-consistent if and only if $\pi \models \Gamma$.*

This leads to a simple characterisation of $\mathcal{H}(1)$-consistency using the maximal inconsistency base: Γ is $\mathcal{H}(1)$-consistent if and only if no inconsistency base involves any strict element of Γ.

Lemma 6.22. *Write MIB$(\Gamma, \mathcal{V})$ as $(\Gamma^{\perp}, C^{\perp})$. Γ is $\mathcal{H}(1)$-consistent if and only if $\Gamma^{\perp} \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$, which is if and only if $\Gamma^{\perp}$ is $\mathcal{H}(1)$-consistent. If Γ is $\mathcal{H}(1)$-inconsistent, then there exists a finite set $\Gamma' \subseteq \Gamma^{\perp}$ such that Γ' is $\mathcal{H}(1)$-inconsistent, and $(\Gamma', C^{\perp})$ is an inconsistency base for $(\Gamma, \mathcal{V})$.*

Proof. Let $\Gamma_{>} = \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}$. First, suppose that Γ is $\mathcal{H}(1)$-consistent. Then, by Corollary 6.21, any maximal Γ-allowed sequence π satisfies Γ. By Proposition 6.15, $\Gamma_{>} \subseteq \mathit{Supp}(\pi)$, and thus, $\Gamma_{>} \subseteq \Gamma \setminus \Gamma^{\perp}$, by Proposition 6.19. Hence, $\Gamma_{>} \cap \Gamma^{\perp} = \emptyset$, and so $\Gamma^{\perp} \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$.

Conversely, suppose that $\Gamma^{\perp} \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$. Proposition 6.19 implies that for any maximal Γ-allowed sequence π, $\Gamma \setminus \Gamma^{\perp} = \mathit{Supp}(\pi)$ and thus, $\Gamma_{>} \subseteq \mathit{Supp}(\pi)$. Proposition 6.15 then implies that $\pi \models \Gamma$, and so Γ is $\mathcal{H}(1)$-consistent.

If Γ is $\mathcal{H}(1)$-consistent, then $\Gamma^{\perp}$ is $\mathcal{H}(1)$-consistent, since $\Gamma^{\perp} \subseteq \Gamma$. Conversely, suppose that $\Gamma^{\perp}$ is $\mathcal{H}(1)$-consistent. Lemma 6.5 implies that $(\Gamma^{\perp}, C^{\perp})$ is an inconsistency base for $(\Gamma^{\perp}, \mathcal{V})$. Proposition 6.8 implies that $\Gamma^{\perp} \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$, which by the first part, implies that Γ is $\mathcal{H}(1)$-consistent.

Now suppose that Γ is $\mathcal{H}(1)$-inconsistent. The first part implies that $\Gamma^{\perp}$ contains a strict statement. By Lemma 6.5(i), there exists finite $\Gamma' \subseteq \Gamma^{\perp}$ such that (Γ', C') is an inconsistency base for $(\Gamma, \mathcal{V})$, and Γ' contains a strict statement. By Lemma 6.5(ii), (Γ', C') is an inconsistency base for $(\Gamma', \mathcal{V})$, and thus, by Proposition 6.8, Γ' is $\mathcal{H}(1)$-inconsistent, since it contains a strict statement. □

The following result shows that this kind of preference inference is compact.

Lemma 6.23. *Consider any* $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ *and* $\varphi \in \mathcal{L}^{\mathcal{A}}$.

(i) If Γ *is* $\mathcal{H}(1)$*-inconsistent, then there exists finite* $\Gamma' \subseteq \Gamma$ *which is* $\mathcal{H}(1)$*-inconsistent.*

(ii) If $\Gamma \models_{\mathcal{H}(1)} \varphi$*, then there exists finite* $\Gamma' \subseteq \Gamma$ *such that* $\Gamma' \models_{\mathcal{H}(1)} \varphi$.

Proof. (i) Suppose that Γ is $\mathcal{H}(1)$-inconsistent. The last part of Lemma 6.22 implies that then there exists finite $\Gamma' \subseteq \Gamma$ which is $\mathcal{H}(1)$-inconsistent.

(ii) Suppose that $\Gamma \models_{\mathcal{H}(1)} \varphi$. Then $\Gamma \cup \{\neg\varphi\}$ is $\mathcal{H}(1)$-inconsistent, by Proposition 6.2. Part (i) implies that there exists finite $\mathcal{H}(1)$-inconsistent $\Delta \subseteq \Gamma \cup \{\neg\varphi\}$. If $\Delta \subseteq \Gamma$, then we can let $\Gamma' = \Delta$, since trivially $\Delta \models_{\mathcal{H}(1)} \varphi$. Otherwise, $\Delta \ni \{\neg\varphi\}$, and we let $\Gamma' = \Delta \setminus \{\neg\varphi\}$. We have $\Gamma' \subseteq \Gamma$, and $\Gamma' \models_{\mathcal{H}(1)} \varphi$, again by Proposition 6.2. □

6.3.3 The Algorithm

The idea behind the algorithm is to build up a maximal $\Gamma^{(\geq)}$-satisfying sequence by repeatedly adding variables to the end as described in the general method for solving the Consistency Problem in Section 4.3.1. Note that an obvious variable mapping for models $\pi \in \mathcal{H}(1)$ is given by $\sigma(\pi)$. We specify in which way these extensions can be selected efficiently. Suppose that we have picked a sequence C' of variables in $\mathcal{V}$ so far. Next, we need to choose a variable c such that, if c opposes some φ in Γ, then φ is already supported by some variable in C' (or else the generated sequence will not satisfy φ).

π is initialised as the empty sequence $()$, which is a minimum model for composition $\circ$ on $\mathcal{H}(1)$. $\pi \leftarrow \pi + c$ means variable c is added to the end of π.

Algorithm 6.1: $\mathcal{H}(1)$-Consistency for Statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$

```
1. π ← ()
2. WHILE ( ∃ c ∈ V \ σ(π) : Opp(c) ⊆ Supp(π) )
3.         Choose some such c
4.         π ← π + c
5. IF   ( π ⊨ Γ ) THEN
6.       RETURN π & "Γ is consistent" and STOP.
7. ELSE RETURN π & "Γ is inconsistent" and STOP.
```

Note that at each stage an element of $\mathit{Next}_\Gamma(\sigma(\pi))$ is chosen, so at each stage π is a Γ-allowed sequence. Also, the termination condition is equivalent to $\mathit{Next}_\Gamma(\sigma(\pi)) = \emptyset$, which implies that the returned π is a maximal Γ-allowed sequence.

The algorithm involves often non-unique choices. However, if we wish, the choosing of c in line 3 can be done based on an ordering $c_1, \ldots, c_m$ of $\mathcal{V}$, where, if there exists more than one $c \in \mathcal{V} \setminus \sigma(\pi)$ such that $\mathit{Opp}(c) \subseteq \mathit{Supp}(\pi)$, we choose the element c_i fulfilling this condition that has smallest index i. The algorithm then becomes deterministic with a unique result following from the given inputs.

A straight-forward implementation runs in $O(|\Gamma||\mathcal{V}|^2)$ time; however, a more careful implementation runs in $O(|\Gamma||\mathcal{V}|)$ time, which we now describe. Let π_k be the lexicographic model after the k-th iteration of the for-loop. In every iteration of the for-loop, we update sets $\mathit{Opp}^\Delta_k(c) = \mathit{Opp}(c) \setminus \mathit{Supp}(\pi_k)$ and $\mathit{Supp}^\Delta_k(c) = \mathit{Supp}(c) \setminus \mathit{Supp}(\pi_k)$ for all $c \in \mathcal{V} \setminus \sigma(\pi_k)$. This costs us $O(|\mathcal{V}\setminus\sigma(\pi_k)|\times|\mathit{Supp}(\pi_k)\setminus\mathit{Supp}(\pi_{k-1})|) = O(|\mathcal{V}\setminus\sigma(\pi_k)|\times|\mathit{Supp}^\Delta_{k-1}(c_k)|)$ more time for every iteration k in which we add variable c_k to π_{k-1}. However, the choice of the next variable c_k be performed in constant time by marking variables c with $\mathit{Opp}^\Delta_{k-1}(c) = \emptyset$. Suppose the algorithm stops after $1 \leq l \leq |\mathcal{V}|$ iterations. Since all $\mathit{Supp}^\Delta_{k-1}(c_k)$ are disjoint, $\sum_{k=1}^{l} |\mathit{Supp}^\Delta_{k-1}(c_k)| = |\mathit{Supp}(\pi_l)| \leq |\Gamma|$. Altogether, the running time is $O(\sum_{k=1}^{l} |\mathcal{V} \setminus \sigma(\pi_k)| \times |\mathit{Supp}^\Delta_{k-1}(c_k)|) \leq O(|\mathcal{V}| \times \sum_{k=1}^{l} |\mathit{Supp}^\Delta_{k-1}(c_k)|)$, and thus the running time is $O(|\mathcal{V}| \times |\Gamma|)$.

Properties of the Algorithm

The algorithm will always generate a lexicographic model satisfying Γ if Γ is $\mathcal{H}(1)$-consistent. It can also be used for computing the maximal inconsistency base. The following result sums up some properties related to the algorithm.

Theorem 6.1: Correctness of the Algorithm

Let π be a sequence returned by the algorithm with inputs Γ and $\mathcal{V}$, and write $\mathit{MIB}(\Gamma, \mathcal{V})$ as $(\Gamma^\perp, C^\perp)$. Then $C^\perp = \mathcal{V} \setminus \sigma(\pi)$ (i.e., the variables that don't appear in π), and $\Gamma^\perp = \Gamma \setminus \mathit{Supp}(\pi)$. We have that $\pi \models \Gamma^{(\geq)}$. Also, Γ is $\mathcal{H}(1)$-consistent if and only if $\mathit{Supp}(\pi)$ contains all the strict elements of Γ, which is if and only if $\Gamma^\perp \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$. If Γ is $\mathcal{H}(1)$-consistent, then $\pi \models \Gamma$.

Proof. By the construction of the algorithm, π is a maximal Γ-allowed sequence, as observed earlier. Proposition 6.19 implies that $C^{\perp} = \mathcal{V} \setminus \sigma(\pi)$ and $\Gamma^{\perp} = \Gamma \setminus \mathit{Supp}(\pi)$. By Proposition 6.11, we have $\pi \models \Gamma^{(\geq)}$. Lemma 6.22 implies that Γ is $\mathcal{H}(1)$-consistent if and only if $\Gamma^{\perp} \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$. Corollary 6.21 implies that Γ is $\mathcal{H}(1)$-consistent if and only if $\pi \models \Gamma$. Proposition 6.15 implies that $\pi \models \Gamma$ if and only if $\mathit{Supp}(\pi)$ contains all the strict elements of Γ. □

The algorithm therefore determines $\mathcal{H}(1)$-consistency, and hence $\mathcal{H}(1)$-deduction (because of Proposition 6.2), in polynomial time, and also generates the maximal inconsistency base.

6.3.4 The case of inconsistent Γ

For the case when Γ is not $\mathcal{H}(1)$-consistent, the output π of the algorithm is a model which, in a sense, comes closest to satisfying Γ: π always satisfies $\Gamma^{(\geq)}$, the non-strict version of Γ, and if any model $\pi' \in \mathcal{H}(1)$ satisfies $\Gamma^{(\geq)}$ and any element φ of Γ, then π also satisfies φ.

Proposition 6.24. *Let π be a sequence returned by the algorithm with inputs Γ and $\mathcal{V}$, and suppose that $\pi' \in \mathcal{H}(1)$ is such that $\pi' \models \Gamma^{(\geq)}$. Then, for all $\varphi \in \Gamma$, if $\pi' \models \varphi$ then $\pi \models \varphi$.*

Proof. Since π is a maximal Γ-allowed sequence, we have (by Proposition 6.11) that $\pi \models \Gamma^{(\geq)}$. Suppose that $\pi' \in \mathcal{H}(1)$ is such that $\pi' \models \Gamma^{(\geq)}$. Proposition 6.20 implies that if $\pi' \models \varphi$ then $\pi \models \varphi$. □

These properties suggest the following way of reasoning with $\mathcal{H}(1)$-inconsistent Γ. Let us define Γ' to be $(\Gamma \setminus \Gamma^{\perp}) \cup \Gamma^{(\geq)}$, where $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^{\perp}, C^{\perp})$. By Theorem 6.1, this is equal to $\mathit{Supp}(\pi) \cup \Gamma^{(\geq)}$, where π is a model generated by the algorithm, enabling easy computation of Γ'. Γ' is $\mathcal{H}(1)$-consistent, since it is satisfied by π. We might then (re-)define the (non-monotonic) deductions from $\mathcal{H}(1)$-inconsistent Γ to be the deductions from Γ'.

6.3.5 Orderings on variables

The preference logic defined here is closely related to a logic based on disjunctive ordering statements. Given a set of variables $\mathcal{V}$, we consider the set of

statements $\mathcal{O}_\mathcal{V}$ of the form $C_1 > C_2$, and of $C_1 \geq C_2$, where C_1 and C_2 are disjoint subsets of $\mathcal{V}$.

We say that $\pi \models C_1 > C_2$, if some variable in C_1 appears in π before every element of C_2, that is, there exists some element of C_1 in π (i.e., $C_1 \cap \sigma(\pi) \neq \emptyset$) and the earliest element of $C_1 \cup C_2$ to appear in π is in C_1.

We say that $\pi \models C_1 \geq C_2$ if either $\pi \models C_1 > C_2$ or no element of C_1 or C_2 appears in π: $(C_1 \cup C_2) \cap \sigma(\pi) = \emptyset$. By Lemma 6.3 we have that

$\pi \models \alpha_\varphi > \beta_\varphi \iff \pi \models \mathit{Supp}^\varphi > \mathit{Opp}^\varphi$,

and $\pi \models \alpha_\varphi \geq \beta_\varphi \iff \pi \models \mathit{Supp}^\varphi \geq \mathit{Opp}^\varphi$.

This shows that the language $\mathcal{O}_\mathcal{V}$ can express anything that can be expressed in $\mathcal{L}^\mathcal{A}$. It can be shown, conversely, that for any statement τ in $\mathcal{O}_\mathcal{V}$, one can define α_φ and β_φ, such that for all $\pi \in \mathcal{H}(1)$, $\pi \models \tau$ if and only if $\pi \models \varphi$ (where φ is strict if and only if τ is strict). Assume that for every variable c, there exist two values l_c and u_c in it's domain $\underline{c}$ such that $l_c < u_c$. (This is a reasonable assumption, as variables that can only assume one value are irrelevant for any decision problem.) Now if τ is, for example, the statement $C_1 > C_2$, we can define $\alpha_\varphi(c) = u_c$ for all $c \in C_2$, and $\alpha_\varphi(c) = l_c$ for $c \in \mathcal{V} \setminus C_2$; and define $\beta_\varphi(c) = u_c$ for all $c \in C_1$, and $\beta_\varphi(c) = l_c$ for $c \in \mathcal{V} \setminus C_1$.

The algorithm can be adapted in the obvious way to the case where we have Γ consisting of (or including) elements in $\mathcal{O}_\mathcal{V}$. When viewed in this way, the algorithm can be seen as a simple extension of a topological sort algorithm; the standard case corresponds to when the ordering statements only involve singleton sets.

6.4 Strong Consistency and $\models^{max}$-Inference

In the set of models $\mathcal{H}(1)$, we allow models involving any subset of $\mathcal{V}$, the set of variables. We could alternatively consider a semantics where we only allow models π that involve all elements of $\mathcal{V}$, i.e., with $\sigma(\pi) = \mathcal{V}$.

In applications, where we can assume that all variables are reflecting features of the alternatives that are relevant for the user, we can consider consistency and inference for preferences based on models that are *complete*, i.e., involve all variables. These models can lead to different inference than the set of mod-

els $\mathcal{H}(1)$, as $\mathcal{H}(1)$ includes models on subsets of variables and even the empty model. However, restricting our considerations to only complete models makes it more likely that preference statements are inconsistent.

Definition 6.4: Strong $\mathcal{H}(1)$-Consistency

Let $\mathcal{H}(1^*)$ be the set of models π of $\mathcal{H}(1)$ with $\sigma(\pi) = \mathcal{V}$. Γ is defined to be *strongly $\mathcal{H}(1)$-consistent* if and only if there exists a model $\pi \in \mathcal{H}(1^*)$ such that $\pi \models \Gamma$.

Let $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^{\perp}, C^{\perp})$. Proposition 6.4 implies that, if Γ is strongly $\mathcal{H}(1)$-consistent then $C^{\perp}$ is empty, and $\Gamma^{\perp}$ consists of all the elements of Γ that are indifferent to all of $\mathcal{V}$, i.e., the set of $\varphi \in \Gamma$ such that $\alpha_{\varphi}(c) = \beta_{\varphi}(c)$ for all $c \in \mathcal{V}$.

There is an associated preference inference based on this restricted set of models. We write $\Gamma \models_{\mathcal{H}(1^*)} \varphi$ if $\pi \models \varphi$ holds for every $\pi \in \mathcal{H}(1^*)$ such that $\pi \models \Gamma$.

This form of deduction can be expressed in terms of strong consistency, as the following result shows.

Lemma 6.25. *If Γ is strongly $\mathcal{H}(1)$-consistent, then $\Gamma \models_{\mathcal{H}(1^*)} \varphi$ holds if and only if $\Gamma \cup \{\neg\varphi\}$ is not strongly $\mathcal{H}(1)$-consistent.*

Proof. First suppose that $\Gamma \cup \{\neg\varphi\}$ is strongly $\mathcal{H}(1)$-consistent. Then there exists $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma \cup \{\neg\varphi\}$ and $\sigma(\pi) = \mathcal{V}$. Thus $\pi \models \Gamma$ and $\pi \not\models \varphi$, showing that $\Gamma \not\models_{\mathcal{H}(1^*)} \varphi$.

Now suppose that $\Gamma \not\models_{\mathcal{H}(1^*)} \varphi$. Then there exists $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$ and $\sigma(\pi) = \mathcal{V}$ and $\pi \not\models \varphi$. Then $\pi \models \Gamma \cup \{\neg\varphi\}$, so $\Gamma \cup \{\neg\varphi\}$ is strongly $\mathcal{H}(1)$-consistent. □

In the next section we will consider a related form of preference inference, where we only consider maximal models.

6.4.1 Max-model inference

For $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, let $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$ be the set of maximal models of Γ within $\mathcal{H}(1)$, i.e., the set of $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$, and for all $\pi' \in \mathcal{H}(1)$ extending π, $\pi' \not\models \Gamma$. Recall the definition of the max-model inference relation $\models^{max}$ from Definition 4.16:

$\Gamma \models^{max} \varphi$ if and only if $\pi \models \varphi$ for all $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$.

As shown in Proposition 4.27 the maximal $\models^*$-models of Γ (i.e., the maximal Γ-allowed sequences, by Proposition 6.11) involve the same set of variables which, by Proposition 6.19, are $\mathcal{V} \setminus C^\perp$, where $\mathit{MIB}(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$. By Theorem 4.1, if Γ is consistent, the set of maximal $\models^*$-models of Γ is the same as the set of maximal models of Γ, and thus the latter also involve the same set of variables $\mathcal{V} \setminus C^\perp$.

The next result shows that the same non-strict preference statements are inferred for the max-model inference relation $\models^{max}$ as for the inference relation $\models_{\mathcal{H}(1)}$.

Proposition 6.26. *Consider any $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, and any preference statement $\alpha \geq \beta$ in $\mathcal{L}^{\mathcal{A}}$.*

(i) Γ is $\mathcal{H}(1)$-consistent if and only if $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma) \neq \emptyset$.

(ii) $\Gamma \models^{max} \alpha \geq \beta \iff \Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$.

Proof. (i) follows easily: If Γ is $\mathcal{H}(1)$-consistent, then there exists some $\pi \in \mathcal{H}(1)$ with $\pi \models \Gamma$, so there exists $\pi' \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$ extending or equalling π. The converse is immediate: If there exists $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$, then $\pi \in \mathcal{H}(1)$ and $\pi \models \Gamma$, so Γ is $\mathcal{H}(1)$-consistent.

(ii) If Γ is not $\mathcal{H}(1)$-consistent, then by part (i), $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma) = \emptyset$, so $\Gamma \models^{max} \alpha \geq \beta$ and $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$ both hold vacuously. Let us thus now assume that Γ is $\mathcal{H}(1)$-consistent.

$\Rightarrow$: Assume $\Gamma \models^{max} \alpha \geq \beta$, and consider any $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$. We need to show that $\pi \models \alpha \geq \beta$. Since $\pi \models \Gamma$, we have $\pi \models \Gamma^{(\geq)}$, and so π is a Γ-allowed model, by Proposition 6.11. Choose, by Lemma 6.17, any maximal Γ-allowed sequence π' extending or equalling π, and we have $\pi' \models \Gamma$. By, Theorem 4.1 and Proposition 6.11, $\pi' \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. Then, $\Gamma \models^{max} \alpha \geq \beta$ implies that $\pi' \models \alpha \geq \beta$. Since $\pi' \sqsupseteq \pi$, $\pi \models^* \alpha \geq \beta$, and by Proposition 6.13, $\pi \models \alpha \geq \beta$.

$\Leftarrow$: Assume $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$, and consider any $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. This implies that $\pi \in \mathcal{H}(1)$ and $\pi \models \Gamma$, so $\pi \models \alpha \geq \beta$ showing that $\Gamma \models^{max} \alpha \geq \beta$. □

In the following, we write $\Gamma \models_{\mathcal{H}(1)} \alpha \equiv \beta$ as an abbreviation of the conjunction of $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$ and $\Gamma \models_{\mathcal{H}(1)} \beta \geq \alpha$; and similarly for other inference relations.

The last result can be used to prove that inferred equivalences are the same for max-model inference, and have a simple form.

Proposition 6.27. *Consider any $\mathcal{H}(1)$-consistent $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. Let MIB$(\Gamma, \mathcal{V})$ equal $(\Gamma^{\perp}, C^{\perp})$. Consider any $\alpha, \beta \in \mathcal{A}$. Then, $\Gamma \models_{\mathcal{H}(1)} \alpha \equiv \beta$ if and only if $\Gamma \models^{max} \alpha \equiv \beta$ if and only if for all $c \in \mathcal{V} \setminus C^{\perp}$, $\alpha(c) = \beta(c)$.*

Proof. First assume that $\Gamma \models_{\mathcal{H}(1)} \alpha \equiv \beta$. This trivially implies that $\Gamma \models^{max} \alpha \equiv \beta$, since every maximal model π ins also in $\mathcal{H}(1)$ and thus $\pi \models_{\mathcal{H}(1)} \alpha \equiv \beta$.

Now assume that $\Gamma \models^{max} \alpha \equiv \beta$. Γ is $\mathcal{H}(1)$-consistent so $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma) \neq \emptyset$, by Proposition 6.26(i). Consider any $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. Then $\alpha \equiv_{\pi} \beta$, which implies that for all $c \in \sigma(\pi)$, $\alpha(c) = \beta(c)$, and thus, by Proposition 6.19, for all $c \in \mathcal{V} \setminus C^{\perp}$, $\alpha(c) = \beta(c)$.

Finally, let us assume that for all $c \in \mathcal{V} \setminus C^{\perp}$, $\alpha(c) = \beta(c)$. Consider any $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$. Proposition 6.4 implies that $\sigma(\pi) \cap C^{\perp} = \emptyset$, i.e., $\sigma(\pi) \subseteq \mathcal{V} \setminus C^{\perp}$. So, for all $c \in \sigma(\pi)$, $\alpha(c) = \beta(c)$, and thus $\alpha \equiv_{\pi} \beta$, and hence $\Gamma \models_{\mathcal{H}(1)} \alpha \equiv \beta$. This completes the proof that the three statements are equivalent. □

The following result shows that the strict inferences with $\models^{max}$ are closely tied with the non-strict inferences.

Proposition 6.28. *$\Gamma \models^{max} \alpha \geq \beta$ if and only if either $\Gamma \models^{max} \alpha \equiv \beta$ or $\Gamma \models^{max} \alpha > \beta$. Also, if Γ is $\mathcal{H}(1)$-consistent, then $\Gamma \models^{max} \alpha > \beta$ holds if and only if $\Gamma \models^{max} \alpha \geq \beta$ and $\Gamma \not\models^{max} \alpha \equiv \beta$.*

Proof. If Γ is not $\mathcal{H}(1)$-consistent, then, by Proposition 6.26(i), $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma) = \emptyset$, so $\Gamma \models^{max} \alpha \geq \beta$ and $\Gamma \models^{max} \alpha \equiv \beta$ (and $\Gamma \models^{max} \alpha > \beta$) hold vacuously, and therefore the equivalence holds. Let us thus now assume that Γ is $\mathcal{H}(1)$-consistent. One direction holds easily: Suppose that $\Gamma \models^{max} \alpha \equiv \beta$ or $\Gamma \models^{max} \alpha > \beta$, and consider any $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. We have either $\pi \models \alpha \equiv \beta$ or $\pi \models \alpha > \beta$, so $\pi \models \alpha \geq \beta$, showing that $\Gamma \models^{max} \alpha \geq \beta$.

Now, let us assume that $\Gamma \models^{max} \alpha \geq \beta$, and that it is not the case that $\Gamma \models^{max} \alpha \equiv \beta$. It is sufficient to show that $\Gamma \models^{max} \alpha > \beta$. Consider any $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. Since, $\Gamma \models^{max} \alpha \geq \beta$, we have $\pi \models \alpha \geq \beta$. Since $\Gamma \not\models^{max} \alpha \equiv \beta$, Proposition 6.27 implies that there exists $c \in \mathcal{V} \setminus C^{\perp}$ such that $\alpha(c) \neq \beta(c)$, where *MIB*$(\Gamma, \mathcal{V}) = (\Gamma^{\perp}, C^{\perp})$. By Proposition 6.19, $\sigma(\pi) = \mathcal{V} \setminus C^{\perp}$, so there exists some $c \in \sigma(\pi)$ such that $\alpha(c) \neq \beta(c)$; let c be earliest such element in

$\sigma(\pi)$. Since $\pi \models \alpha \geq \beta$, we have $\alpha(c) > \beta(c)$, so $\pi \models \alpha > \beta$. This shows that $\Gamma \models^{max} \alpha > \beta$, as required.

Assume that Γ is $\mathcal{H}(1)$-consistent. Suppose that $\Gamma \models^{max} \alpha > \beta$ holds. Then clearly, $\Gamma \models^{max} \alpha \geq \beta$. Consider any $\pi \models \Gamma$. Then we have $\alpha \succ_\pi \beta$, so we do not have $\alpha \equiv_\pi \beta$, which implies that $\Gamma \models_{\mathcal{H}(1)} \alpha \equiv \beta$ does not hold. Conversely, suppose that $\Gamma \models^{max} \alpha \geq \beta$ and $\Gamma \not\models^{max} \alpha \equiv \beta$. The first part then implies that $\Gamma \models^{max} \alpha > \beta$. □

6.4.2 Properties of strong consistency and the associated inference

The following result shows that the consequences of Γ with respect to $\models_{\mathcal{H}(1^*)}$ are the same as those with respect to $\models^{max}$, when Γ is strongly $\mathcal{H}(1)$-consistent. (Of course, if Γ is not strongly $\mathcal{H}(1)$-consistent then all φ in $\mathcal{L}^{\mathcal{A}}$ are consequences of $\models_{\mathcal{H}(1^*)}$.)

Lemma 6.29. *If Γ is strongly $\mathcal{H}(1)$-consistent, then, for any $\varphi \in \mathcal{L}^{\mathcal{A}}$, $\Gamma \models_{\mathcal{H}(1^*)} \varphi \iff \Gamma \models^{max} \varphi$.*

Proof. Assume that Γ is strongly $\mathcal{H}(1)$-consistent, so there exists a model π' with $\sigma(\pi') = \mathcal{V}$. By definition of $\models_{\mathcal{H}(1^*)}$ and $\models^{max}$ it is sufficient to show that $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$ is equal to the set $\mathcal{H}$ of all $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$ and $\sigma(\pi) = \mathcal{V}$. It immediately follows that $\mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma) \supseteq \mathcal{H}$. Conversely, consider any $\pi \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. Since $\pi' \in \mathcal{H}$, we have $\pi' \in \mathcal{M}^{\max}_{\mathcal{H}(1)}(\Gamma)$. Proposition 6.19 implies that $\sigma(\pi) = \sigma(\pi') = \mathcal{V}$, proving that $\pi \in \mathcal{H}$. □

The next discussion shows that the non-strict $\models_{\mathcal{H}(1^*)}$ inferences are the same as the non-strict $\models_{\mathcal{H}(1)}$ inferences, and that (in contrast to the case of $\models_{\mathcal{H}(1)}$), the strict $\models_{\mathcal{H}(1^*)}$ inferences almost correspond with the non-strict ones. This also implies that the algorithm in Section 6.3.3 can be used to efficiently determine the $\models_{\mathcal{H}(1^*)}$ inferences.

To illustrate the difference between the $\models_{\mathcal{H}(1)}$ inferences and the $\models_{\mathcal{H}(1^*)}$ inferences for the case of strict statements, consider some strongly $\mathcal{H}(1)$-consistent Γ which only includes non-strict statements. Then, for every strict preference statement $\alpha > \beta$, we will have $\Gamma \not\models_{\mathcal{H}(1)} \alpha > \beta$ since the empty sequence satisfies Γ but not $\alpha > \beta$. However, we will have $\Gamma \models_{\mathcal{H}(1^*)} \alpha > \beta$ if $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$

and $\Gamma \not\models_{\mathcal{H}(1)} \beta \geq \alpha$. For example, if Γ is just $\{\alpha \geq \beta\}$, where for some $c \in \mathcal{V}$, $\alpha(c) > \beta(c)$, then we will have $\Gamma \models_{\mathcal{H}(1^*)} \alpha > \beta$ but not $\Gamma \models_{\mathcal{H}(1)} \alpha > \beta$.

Proposition 6.30. *Let MIB$(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$. $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ is strongly $\mathcal{H}(1)$-consistent if and only if $C^\perp = \emptyset$ and $\Gamma \cap \mathcal{L}^{\mathcal{A}}_{>} \subseteq$ Supp$(\mathcal{V})$, where Supp$(\mathcal{V})$ is the set of statements $\varphi \in \Gamma$ that are supported by some variable $c \in \mathcal{V}$.*

Suppose that $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ is strongly $\mathcal{H}(1)$-consistent. Then,

(i) $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta \iff \Gamma \models_{\mathcal{H}(1^*)} \alpha \geq \beta$*;*

(ii) $\Gamma \models_{\mathcal{H}(1^*)} \alpha \equiv \beta$ *if and only if α and β agree on all of $\mathcal{V}$, i.e., for all $c \in \mathcal{V}$, $\alpha(c) = \beta(c)$;*

(iii) $\Gamma \models_{\mathcal{H}(1^*)} \alpha > \beta$ *if and only if* $\Gamma \models_{\mathcal{H}(1)} \alpha \geq \beta$ *and α and β differ on some element of $\mathcal{V}$, i.e., there exists $c \in \mathcal{V}$ such that $\alpha(c) \neq \beta(c)$.*

Proof. First, suppose that Γ is strongly $\mathcal{H}(1)$-consistent. Then there exists $\pi' \in \mathcal{H}(1)$ such that $\pi' \models \Gamma$ and $\sigma(\pi') = \mathcal{V}$. Since $\pi' \models \Gamma^{(\geq)}$, by Proposition 6.11, π' is a Γ-allowed sequence. By Lemma 6.17, there exists a maximal Γ-allowed sequence π extending or equalling π', so, since $\sigma(\pi') = \mathcal{V}$, we must have $\pi = \pi'$. Proposition 6.19 implies that $C^\perp = \emptyset$ and $\Gamma^\perp = \Gamma \setminus$ Supp$(\pi) = \Gamma \setminus$ Supp$(\mathcal{V})$, and Lemma 6.22 shows then that $(\Gamma \setminus$ Supp$(\mathcal{V})) \cap \mathcal{L}^{\mathcal{A}}_{>} = \emptyset$, which implies that $\Gamma \cap \mathcal{L}^{\mathcal{A}}_{>} \subseteq$ Supp$(\mathcal{V})$.

Conversely, suppose that $C^\perp = \emptyset$ and $\Gamma \cap \mathcal{L}^{\mathcal{A}}_{>} \subseteq$ Supp$(\mathcal{V})$. Let π be a maximal Γ-allowed sequence. Proposition 6.19 implies that $\sigma(\pi) = \mathcal{V}$. Then Supp$(\pi) =$ Supp$(\mathcal{V})$, and Proposition 6.15 implies that $\pi \models \Gamma$, showing that Γ is strongly $\mathcal{H}(1)$-consistent.

Now suppose that Γ is strongly $\mathcal{H}(1)$-consistent. Lemma 6.29 implies that for any $\varphi \in \mathcal{L}^{\mathcal{A}}$, $\Gamma \models_{\mathcal{H}(1^*)} \varphi \iff \Gamma \models^{max} \varphi$. Part (i) then follows by Proposition 6.26(ii). Part (ii) follows from Proposition 6.27, using the fact that $C^\perp$ is empty. Part (iii) follows from part (ii) and Proposition 6.28. □

The next result shows that $\models_{\mathcal{H}(1)}$ inference is not affected if one removes the variables in the MIB.

Proposition 6.31. *Suppose that Γ is $\mathcal{H}(1)$-consistent, let MIB$(\Gamma, \mathcal{V}) = (\Gamma^\perp, C^\perp)$, and let $\mathcal{H}'(1)$ be the lexicographic models in $\mathcal{H}(1)$ only involving variables $\mathcal{V} \setminus C^\perp$. Then Γ is strongly $\mathcal{H}'(1)$-consistent, and $\Gamma \models_{\mathcal{H}(1)} \varphi$ if and only if $\Gamma \models_{\mathcal{H}'(1)} \varphi$.*

Proof. By Theorem 6.1, any output of the algorithm is in $C'(1^*)$ and satisfies Γ. Thus Γ is strongly $\mathcal{H}'(1)$-consistent. Let $H' = \{\pi \in \mathcal{H}'(1) \ : \ \pi \models \Gamma\}$ and $H = \{\pi \in \mathcal{H}(1) \ : \ \pi \models \Gamma\}$. Then $H' \subseteq H$, because $\mathcal{H}'(1) \subseteq \mathcal{H}(1)$. By Proposition 6.4, for every $\pi \in H$, we have $\sigma(\pi) \cap C^{\perp} = \emptyset$, and hence $\pi \in H'$. Thus $H' = H$ and $\Gamma \models_{\mathcal{H}(1)} \varphi$ if and only if $\Gamma \models_{\mathcal{H}'(1)} \varphi$. □

6.5 Proof Theory for $\mathcal{H}(1)$-Inference

Preference inference has been defined semantically, and we have an efficient algorithm for the simple lexicographic case. From a logical perspective, it is natural to consider if we can construct an equivalent syntactical definition of inference via a proof theory; this can give another view of the assumptions being made by the logic. In this section, we construct such a proof theory for preference inference based on simple lexicographic models, involving an axiom schema and a number of fairly simple inference rules. We consider a fixed set of variables $\mathcal{V}$ for which variable values are real numbers so that subtraction and multiplication with scalars are well defined. We abbreviate $\models_{\mathcal{H}(1)}$ to just $\models$.

We make use of a form of Pareto (point wise) ordering on alternatives, and we define a kind of addition and rescaling operation on alternatives and thus on preference statements.

We define the following well-known point wise (or weak) Pareto ordering on alternatives. For $\alpha, \beta \in \mathcal{A}$, $\alpha \succcurlyeq_{par} \beta \iff$ for all $c \in \mathcal{V}$, $\alpha(c) \geq \beta(c)$. We also define the Pareto Difference relation between elements of $\mathcal{L}^{\mathcal{A}}$.

Definition 6.5: Pareto Difference Relation

For $\psi, \theta \in \mathcal{L}^{\mathcal{A}}$, we say that $\psi \succcurlyeq_{parD} \theta$ holds if and only if

(i) ψ and θ are either both strict or both non-strict; and

(ii) for all $c \in \mathcal{V}$, $\beta_{\psi}(c) - \alpha_{\psi}(c) \geq \beta_{\theta}(c) - \alpha_{\theta}(c)$.

Note that the definition of $\psi \succcurlyeq_{parD} \theta$ requires variable domains to be closed under a subtraction operation, which is the case due to our assumption that variable values are subsets of the real numbers. Thus, if $\psi \succcurlyeq_{parD} \theta$ and $\alpha_{\psi}(c) \geq \beta_{\psi}(c)$ then $\alpha_{\theta}(c) \geq \beta_{\theta}(c)$. If $\psi \succcurlyeq_{parD} \theta$ and $\pi \models \psi$, then $\pi \models \theta$ (see Lemma 6.32(vi) below).

Point wise multiplication of alternatives and preference statements: Let F be the set of functions from $\mathcal{V}$ to the strictly positive rational numbers. For $f \in F$, we define $\frac{1}{f} \in F$ in the obvious way: Let $f(c) = \frac{1}{f(c)}$ for $c \in \mathcal{V}$. Let f be an arbitrary element of F.

- For $\alpha, \gamma \in \mathcal{A}$, we say that $\alpha \doteq f\gamma$ if for all $c \in \mathcal{V}$, $\alpha(c) = f(c) \times \gamma(c)$ (where $\times$ is the standard multiplication).
- For $\varphi, \psi \in \mathcal{L}^{\mathcal{A}}$, we say that $\varphi \doteq f\psi$ if (i) $\alpha_\varphi \doteq f\alpha_\psi$ and $\beta_\varphi \doteq f\beta_\psi$, and (ii) φ is strict if and only if ψ is strict.

Note that if $\varphi \doteq f\psi$ then for all $c \in \mathcal{V}$, $\alpha_\varphi(c) \geq \beta_\varphi(c) \iff \alpha_\psi(c) \geq \beta_\psi(c)$. It is then easy to show that if $\pi \in \mathcal{H}(1)$ and $\varphi \doteq f\psi$ then $\pi \models \varphi$ if and only if $\pi \models \psi$ (see Lemma 6.32(iv) below).

Addition of alternatives and preference statements:

- For $\alpha, \beta, \gamma \in \mathcal{A}$, we say that $\gamma \doteq \alpha + \beta$ if for all $c \in \mathcal{V}$, $\gamma(c) = \alpha(c) + \beta(c)$.
- For $\varphi, \psi, \chi \in \mathcal{L}^{\mathcal{A}}$, we say that $\varphi \doteq \psi + \chi$ if (i) $\alpha_\varphi \doteq \alpha_\psi + \alpha_\chi$, and $\beta_\varphi \doteq \beta_\psi + \beta_\chi$; and (ii) φ is non-strict if both ψ and χ are non-strict, and otherwise, φ is strict.

6.5.1 Syntactic deduction $\vdash$ and soundness of inference rules

As usual the proof theory is constructed from axioms and inference rules.

Axioms:

$\alpha \geq \beta$ for all $\alpha, \beta \in \mathcal{A}$ with $\alpha \succcurlyeq_{par} \beta$.

Inference Rules Schemata:

(1) For any $\alpha, \beta \in \mathcal{A}$: *From* $\alpha > \beta$ *deduce* $\alpha \geq \beta$.

[Strict to Non-Strict]

(2) For $\chi \in \mathcal{L}^{\mathcal{A}}$ such that $\chi \doteq \varphi + \psi$: *From* φ *and* ψ *deduce* χ.

[Addition]

(3) For $f \in F$ and $\varphi \in \mathcal{L}^{\mathcal{A}}$ such that $\varphi \doteq f\psi$: *From* ψ *deduce* φ.

[Point wise Multiplication]

(4) For any $\alpha \in \mathcal{A}$ and any $\varphi \in \mathcal{L}^{\mathcal{A}}$: *From* $\alpha > \alpha$ *deduce* φ.

[Inconsistent Statement]

(5) For any $\psi, \theta \in \mathcal{L}^{\mathcal{A}}$ such that $\psi \succcurlyeq_{parD} \theta$: *From ψ deduce θ.*

[Pareto Difference]

Defining syntactic deduction $\vdash$: Let Γ be a subset of $\mathcal{L}^{\mathcal{A}}$ and $\varphi \in \mathcal{A}$. We say that φ can be proven from Γ, written $\Gamma \vdash \varphi$, if there exists a sequence $\varphi_1, \ldots, \varphi_k$ of elements of $\mathcal{L}^{\mathcal{A}}$ such that $\varphi_k = \varphi$ and for all $i = 1, \ldots, k$, either $\varphi_i \in \Gamma$ or φ_i is an axiom, or there exists an instance of one of the inference rules with consequent φ_i and such that the antecedents are in $\{\varphi_1, \ldots, \varphi_{i-1}\}$. Relation $\vdash$ depends strongly on the set of alternatives $\mathcal{A}$; e.g., $\{\varphi, \psi\} \vdash \varphi + \psi$ (if and) only if $\varphi + \psi \in \mathcal{L}^{\mathcal{A}}$, i.e., only if $\alpha_\varphi + \alpha_\psi$ and $\beta_\varphi + \beta_\psi$ are in $\mathcal{A}$. We write $\vdash$ as $\vdash_{\mathcal{A}}$ if we want to emphasise this dependency. It can happen that for $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}} \subseteq \mathcal{L}^{\mathcal{B}}$, we have $\Gamma \vdash_{\mathcal{B}} \varphi$, but $\Gamma \nvdash_{\mathcal{A}} \varphi$. (We could also write $\models_{\mathcal{A}}$ to emphasise the dependency on $\mathcal{A}$; however, it isn't usually important to do so, since for $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}} \subseteq \mathcal{L}^{\mathcal{B}}$, we have $\Gamma \models_{\mathcal{B}} \varphi \iff \Gamma \models_{\mathcal{A}} \varphi$.)

Any given set of alternatives may not be closed under addition (for instance), and there may be $\alpha, \beta \in \mathcal{A}$ with no $\gamma \in \mathcal{A}$ such that $\gamma \doteq \alpha + \beta$. We assume that we can augment $\mathcal{A}$ with additional alternatives, and for any function $g : \mathcal{V} \to \mathbb{Q}^+$, we can construct an alternative α with, for all $c \in \mathcal{V}$, $\alpha(c) = g(c)$.

Next we state a lemma showing soundness of the axioms and inference rules, which is used to prove soundness of the associated syntactic deduction (Proposition 6.33).

Lemma 6.32. *Consider any $\pi \in \mathcal{H}(1)$, any $\alpha, \beta \in \mathcal{A}$, and any $\varphi, \psi, \chi, \theta \in \mathcal{L}^{\mathcal{A}}$.*

(i) If $\alpha \succcurlyeq_{par} \beta$, then $\pi \models \alpha \geq \beta$.

(ii) If $\pi \models \alpha > \beta$, then $\pi \models \alpha \geq \beta$.

(iii) If $\chi \doteq \varphi + \psi$ and $\pi \models \varphi$ and $\pi \models \psi$, then $\pi \models \chi$.

(iv) If $\varphi \doteq f\psi$, then $\pi \models \varphi \iff \pi \models \psi$.

(v) $\pi \not\models \alpha > \alpha$.

(vi) If $\pi \models \psi$ and $\psi \succcurlyeq_{parD} \theta$, then $\pi \models \theta$.

Proof. Write π as $(c_1, \ldots, c_k)$. For $\varphi \in \mathcal{L}^{\mathcal{A}}$ we define i^φ to be $k+1$ if for all $i = 1, \ldots, k$, $\alpha_\varphi(c_i) = \beta_\varphi(c_i)$; otherwise, we define i^φ to be the minimum i such that $\alpha_\varphi(c_i) \neq \beta_\varphi(c_i)$. Then $\alpha_\varphi \equiv_\pi \beta_\varphi \iff i^\varphi = k+1$, and $\pi \models \alpha_\varphi > \beta_\varphi \iff i^\varphi \leq k$ and $\alpha_\varphi(c_{i^\varphi}) > \beta_\varphi(c_{i^\varphi})$.

(i): Assume that $\alpha \succcurlyeq_{par} \beta$, so that for all $c \in \mathcal{V}$, we have $\alpha(c) \geq \beta(c)$. This implies $\alpha \succcurlyeq_{\pi} \beta$ and thus $\pi \models \alpha \geq \beta$.

(ii): Assume that $\pi \models \alpha > \beta$, so that $\alpha \succ_{\pi} \beta$. This implies $\alpha \succcurlyeq_{\pi} \beta$ and hence $\pi \models \alpha \geq \beta$.

(iii): Assume that $\chi \doteq \varphi + \psi$, and $\pi \models \varphi$ and $\pi \models \psi$.

Case (I): $i^{\varphi} = i^{\psi} = k + 1$. Then for all $i = 1, \ldots, k$, $\alpha_{\varphi}(c_i) = \beta_{\varphi}(c_i)$ and $\alpha_{\psi}(c_i) = \beta_{\psi}(c_i)$. Then, $\alpha_{\chi}(c_i) = \alpha_{\varphi}(c_i) + \alpha_{\psi}(c_i) = \beta_{\varphi}(c_i) + \beta_{\psi}(c_i) = \beta_{\chi}(c_i)$, so $i^{\chi} = k + 1$, which implies that $\alpha_{\chi} \equiv_{\pi} \beta_{\chi}$. We have $\alpha_{\varphi} \equiv_{\pi} \beta_{\varphi}$, and also $\pi \models \varphi$, so φ is non-strict. Similarly, ψ is non-strict. Thus χ is non-strict, and so $\pi \models \chi$.

Case (II): $i^{\varphi} = i^{\psi} \leq k$. Because $\alpha_{\varphi}(c_{i^{\varphi}}) \neq \beta_{\varphi}(c_{i^{\varphi}})$ and $\pi \models \varphi$, we have $\alpha_{\varphi}(c_{i^{\varphi}}) > \beta_{\varphi}(c_{i^{\varphi}})$. The same argument implies that $\alpha_{\psi}(c_{i^{\varphi}}) > \beta_{\psi}(c_{i^{\varphi}})$. We then have $\alpha_{\chi}(c_{i^{\varphi}}) > \beta_{\chi}(c_{i^{\varphi}})$, and $i^{\chi} = i^{\varphi}$. This implies that $\pi \models \alpha_{\chi} > \beta_{\chi}$, and thus, $\pi \models \chi$, whether χ is strict or non-strict.

Case (III): $i^{\varphi} < i^{\psi}$. Arguing as in Case (II), we have $\alpha_{\varphi}(c_{i^{\varphi}}) > \beta_{\varphi}(c_{i^{\varphi}})$. We also have $\alpha_{\psi}(c_{i^{\varphi}}) = \beta_{\psi}(c_{i^{\varphi}})$. We then have $\alpha_{\chi}(c_{i^{\varphi}}) > \beta_{\chi}(c_{i^{\varphi}})$, and $i^{\chi} = i^{\varphi}$. Again we have $\pi \models \chi$, whether χ is strict or non-strict.

Case (IV): $i^{\varphi} > i^{\psi}$. This is similar to Case (III), but with the roles of φ and ψ reversed.

(iv): Assume that $\varphi \doteq f\psi$, and consider any $c \in \mathcal{V}$. Because $f(c) > 0$, we have $\alpha_{\varphi}(c) = \beta_{\varphi}(c)$ if and only if $\alpha_{\psi}(c) = \beta_{\psi}(c)$; and $\alpha_{\varphi}(c) > \beta_{\varphi}(c)$ if and only if $\alpha_{\psi}(c) > \beta_{\psi}(c)$. This shows that $\pi \models \varphi \iff \pi \models \psi$.

(v): $\pi \not\models \alpha > \alpha$ follows since $\alpha \equiv_{\pi} \alpha$ and so $\alpha \not\succ_{\pi} \alpha$.

(vi): Suppose that $\pi \models \psi$ and $\psi \succcurlyeq_{parD} \theta$, so that ψ and θ are either both strict or both non-strict; and for all $c \in \mathcal{V}$, $\beta_{\psi}(c) - \alpha_{\psi}(c) \geq \beta_{\theta}(c) - \alpha_{\theta}(c)$. If it were the case that $i^{\psi} < i^{\theta}$, then because $\pi \models \psi$, we would have that $\alpha_{\psi}(c_{i^{\psi}}) > \beta_{\psi}(c_{i^{\psi}})$ and $\alpha_{\theta}(c_{i^{\psi}}) = \beta_{\theta}(c_{i^{\psi}})$, and thus, $\beta_{\psi}(c_{i^{\psi}}) - \alpha_{\psi}(c_{i^{\psi}}) < 0 = \beta_{\theta}(c_{i^{\psi}}) - \alpha_{\theta}(c_{i^{\psi}})$, which contradicts $\psi \succcurlyeq_{parD} \theta$. Thus we must have that $i^{\psi} \geq i^{\theta}$.

First consider the case when $i^{\theta} = k + 1$. Then $i^{\psi} = k + 1$, and so $\alpha_{\theta} \equiv_{\pi} \beta_{\theta}$ and $\alpha_{\psi} \equiv_{\pi} \beta_{\psi}$. The latter implies that ψ is non-strict, since $\pi \models \psi$. Then θ is non-strict and thus, $\pi \models \theta$.

Now consider the case when $i^{\theta} \leq k$, and thus $\alpha_{\theta}(c_{i^{\theta}}) \neq \beta_{\theta}(c_{i^{\theta}})$. We showed earlier that $i^{\theta} \leq i^{\psi}$. If $i^{\theta} = i^{\psi}$, then $\pi \models \psi$ implies that $\alpha_{\psi}(c_{i^{\theta}}) > \beta_{\psi}(c_{i^{\theta}})$. If $i^{\theta} < i^{\psi}$, then $\alpha_{\psi}(c_{i^{\theta}}) = \beta_{\psi}(c_{i^{\theta}})$. So, in either case we have $\alpha_{\psi}(c_{i^{\theta}}) \geq \beta_{\psi}(c_{i^{\theta}})$,

i.e., $\beta_\psi(c_{i^\theta}) - \alpha_\psi(c_{i^\theta}) \leq 0$. The assumption $\psi \succcurlyeq_{parD} \theta$ then implies that $\beta_\theta(c_{i^\theta}) - \alpha_\theta(c_{i^\theta}) \leq 0$, and so, $\alpha_\theta(c_{i^\theta}) \geq \beta_\theta(c_{i^\theta})$. Since $i^\theta \leq k$ we have $\alpha_\theta(c_{i^\theta}) > \beta_\theta(c_{i^\theta})$, showing that $\pi \models \alpha_\theta > \beta_\theta$, and therefore $\pi \models \theta$ whether θ is strict or non-strict. □

We are now ready to state and prove the soundness result.

Proposition 6.33. *For $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}}$, and any $\mathcal{B} \supseteq \mathcal{A}$, if $\Gamma \vdash_{\mathcal{B}} \varphi$ then $\Gamma \models_{\mathcal{A}} \varphi$.*

Proof. First note that if Γ is $\mathcal{H}(1)$-inconsistent, then there is nothing to prove, since $\Gamma \models_{\mathcal{A}} \varphi$ follows trivially. So, let us assume now that Γ is $\mathcal{H}(1)$-consistent. We use an inductive proof based on Lemma 6.32. Suppose that $\Gamma \vdash_{\mathcal{B}} \varphi$. Consider any $\pi \in \mathcal{H}(1)$ such that $\pi \models \Gamma$. We need to show that $\pi \models \varphi$. Since $\Gamma \vdash_{\mathcal{B}} \varphi$ there exists a sequence $\varphi_1, \ldots, \varphi_k$ of elements of $\mathcal{L}^{\mathcal{B}}$ such that $\varphi_k = \varphi$ and for all $i = 1, \ldots, k$, either $\varphi_i \in \Gamma$ or φ_i is an axiom, or there exists an instance of one of the inference rules with consequent φ_i and such that the antecedents are in $\{\varphi_1, \ldots, \varphi_{i-1}\}$. Consider any $i \in \{1, \ldots, k\}$. We will prove that, if for all $j < i$, $\pi \models \varphi_j$, then $\pi \models \varphi_i$. This then implies that for all $i = 1, \ldots, k$, we have $\pi \models \varphi_i$, and thus $\pi \models \varphi_k$, as required.

Therefore, let i be some arbitrary element in $\{1, \ldots, k\}$, and assume that for all $j < i$, $\pi \models \varphi_j$. We will prove that $\pi \models \varphi_i$. Let us abbreviate φ_i to be θ. One of the cases (1)–(7) below applies. We consider each case in turn.

(1): θ equals $\alpha \geq \beta$ for some $\alpha, \beta \in \mathcal{B}$, and there exists some $j < i$ with φ_j equalling $\alpha > \beta$. Since $\pi \models \varphi_j$, by Lemma 6.32(ii), we have $\pi \models \alpha \geq \beta$, i.e., $\pi \models \theta$.

(2): θ equals χ for some $\chi \in \mathcal{L}^{\mathcal{B}}$ such that $\chi \doteq \varphi + \psi$, and for some $j, l < i$ we have $\varphi = \varphi_j$ and $\psi = \varphi_l$. Since $\pi \models \varphi_j, \varphi_l$, Lemma 6.32(iii) implies that $\pi \models \theta$.

(3): There exists $j < i$ and $f \in F$ such that $\theta \doteq f\varphi_j$. Lemma 6.32(iv) implies that $\pi \models \theta$.

(4): There exists $\alpha \in \mathcal{B}$ and $j < i$ such that φ_j equals $\alpha > \alpha$, so we have $\pi \models \alpha > \alpha$. However, by Lemma 6.32(v), this is impossible, so Case (4) cannot arise.

(5): There exists $j < i$ such that $\psi = \varphi_j \in \mathcal{L}^{\mathcal{B}}$ and $\psi \succcurlyeq_{parD} \theta$. Lemma 6.32(vi) implies $\pi \models \theta$.

(6): $\theta \in \Gamma$. Then $\pi \models \theta$.

(7): θ is equal to $\alpha \geq \beta$ for some $\alpha, \beta \in \mathcal{B}$ such that $\alpha \succcurlyeq_{par} \beta$. Lemma 6.32(i) implies $\pi \models \theta$.

□

6.5.2 Completeness of Proof Theory

We now give a pair of technical lemmas which we will use in the completeness proof.

Lemma 6.34. *Consider any $\mathcal{H}(1)$-inconsistent $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, and suppose that $(\{\varphi_1, \ldots, \varphi_k\}, C')$ is an inconsistency base for $(\Gamma, \mathcal{V})$, with $\{\varphi_1, \ldots, \varphi_k\}$ being inconsistent. Then there exist strictly positive functions $f_1, \ldots, f_k \in F$, set of alternatives $\mathcal{B} \supseteq \mathcal{A}$ with $\mathcal{B} \setminus \mathcal{A}$ finite, preference statement $\rho \in \mathcal{L}^{\mathcal{B}}$ and strict preference statement ψ in $\mathcal{L}^{\mathcal{B}}$ such that $\rho \doteq f_1\varphi_1 + \cdots + f_{k-1}\varphi_{k-1}$ and $\psi \doteq f_1\varphi_1 + \cdots + f_k\varphi_k$, and $\Gamma \vdash_{\mathcal{B}} \rho$ and $\Gamma \vdash_{\mathcal{B}} \psi$, and $\beta_\psi \succcurlyeq_{par} \alpha_\psi$.*

Proof. Let $T = \{|\alpha_{\varphi_i}(c) - \beta_{\varphi_i}(c)| : c \in \mathcal{V}, i \in \{1, \ldots, k\}\} \setminus \{0\}$. If $T = \emptyset$, then set $a = b = 1$, and if $T \neq \emptyset$ let $a = \min T$ and let $b = \max T$, so $0 < a \leq b$. For $i = 1, \ldots, k$ and $c \in \mathcal{V}$, we define $f_i(c) = 1$ if $\alpha_{\varphi_i}(c) < \beta_{\varphi_i}(c)$, and otherwise, we define $f_i(c) = d$ where $d = a/(kb) > 0$.

For $i = 1, \ldots, k$, we include elements $\gamma_i, \delta_i, \epsilon_i, \lambda_i$ in $\mathcal{B}$, where $\gamma_i \doteq f_i\alpha_{\varphi_i}$, and $\delta_i \doteq f_i\beta_{\varphi_i}$; and we let $\epsilon_1 = \gamma_1$ and $\lambda_1 = \delta_1$, and for $i = 2, \ldots, k$, $\epsilon_i \doteq \epsilon_{i-1} + \gamma_i$, and $\lambda_i \doteq \lambda_{i-1} + \delta_i$.

There exists $\psi_1 \in \mathcal{L}^{\mathcal{B}}$ with $\psi_1 \doteq f_1\varphi_1$, and $\alpha_{\psi_1} = \gamma_1 = \epsilon_1$ and $\beta_{\psi_1} = \delta_1 = \lambda_1$. Similarly, for $i = 2, \ldots, k$, there exists $\psi_i \in \mathcal{L}^{\mathcal{B}}$ with $\psi_i \doteq \psi_{i-1} + f_i\varphi_i$, and $\alpha_{\psi_i} = \epsilon_i$ and $\beta_{\psi_i} = \lambda_i$.

By the Addition and Point wise Multiplication rules, for each $i = 1, \ldots, k$, we have $\Gamma \vdash_{\mathcal{B}} \psi_i$. Abbreviate ψ_k to ψ and ψ_{k-1} to ρ. We have $\Gamma \vdash_{\mathcal{B}} \psi$ and $\psi \doteq f_1\varphi_1 + \cdots + f_k\varphi_k$, and $\Gamma \vdash_{\mathcal{B}} \rho$ and $\rho \doteq f_1\varphi_1 + \cdots + f_{k-1}\varphi_{k-1}$. Since $\{\varphi_1, \ldots, \varphi_k\}$ is inconsistent, some φ_i is strict (else the empty model satisfies them all), and therefore, ψ is a strict preference statement.

Consider any $c \in \mathcal{V} \setminus C'$. By Definition 6.1(i), $\alpha_{\varphi_i}(c) = \beta_{\varphi_i}(c)$ for all $i = 1, \ldots, k$. Thus $\alpha_\psi(c) = \beta_\psi(c)$.

Now consider any $c \in C'$. For any $j \in \{1, \ldots, k\}$, $\alpha_{\varphi_j}(c) - \beta_{\varphi_j}(c) \leq b$, and so $\gamma_j(c) - \delta_j(c) \leq bd = a/k$. By Definition 6.1(ii), there exists some $i \in \{1 \ldots, k\}$

such that $\alpha_{\varphi_i}(c) < \beta_{\varphi_i}(c)$. This implies that $T \neq \emptyset$. We have $\alpha_{\varphi_i}(c) - \beta_{\varphi_i}(c) \leq -a$, and thus $\gamma_i(c) - \delta_i(c) \leq -a (< 0)$. Now, $\alpha_\psi(c) = \sum_{j=1}^{k} \gamma_j(c)$ and $\beta_\psi(c) = \sum_{j=1}^{k} \delta_j(c)$. Therefore, $\alpha_\psi(c) - \beta_\psi(c) \leq -a + (k-1)a/k < 0$. We have shown that for all $c \in \mathcal{V}$, $\alpha_\psi(c) \leq \beta_\psi(c)$, so $\beta_\psi \succcurlyeq_{par} \alpha_\psi$. □

Lemma 6.35. *Suppose $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}}$, and that Γ is $\mathcal{H}(1)$-consistent and $\Gamma \models \varphi$. Then there exists $\mathcal{B} \supseteq \mathcal{A}$ (with $\mathcal{B} \setminus \mathcal{A}$ finite), and $\chi, \theta \in \mathcal{L}^{\mathcal{B}}$ such that $\Gamma \vdash_{\mathcal{B}} \chi$, and θ is strict and $\theta \doteq \chi + \neg\varphi$, and $\beta_\theta \succcurlyeq_{par} \alpha_\theta$.*

Proof. By Lemma 6.2, $\Gamma \cup \{\neg\varphi\}$ is $\mathcal{H}(1)$-inconsistent. By Lemma 6.22 there exists an inconsistency base (Δ, C') for $(\Gamma \cup \{\neg\varphi\}, \mathcal{V})$ with Δ being a finite and $\mathcal{H}(1)$-inconsistent subset of $\Gamma \cup \{\neg\varphi\}$, and $C' \subseteq \mathcal{V}$. Now, Δ contains $\neg\varphi$, since Δ is $\mathcal{H}(1)$-inconsistent and Γ is $\mathcal{H}(1)$-consistent. We write Δ as $\{\varphi_1, \ldots, \varphi_k\}$ with $\varphi_k = \neg\varphi$.

By Lemma 6.34, there exist strictly positive functions $f_1, \ldots, f_k \in F$, set of alternatives $\mathcal{B} \supseteq \mathcal{A}$ with $\mathcal{B} \setminus \mathcal{A}$ finite, preference statement $\rho \in \mathcal{L}^{\mathcal{B}}$ and strict preference statement ψ in $\mathcal{L}^{\mathcal{B}}$ such that $\rho \doteq f_1\varphi_1 + \cdots + f_{k-1}\varphi_{k-1}$ and $\psi \doteq f_1\varphi_1 + \cdots + f_k\varphi_k$, $\Gamma \vdash_{\mathcal{B}} \rho$ and $\Gamma \vdash_{\mathcal{B}} \psi$, and $\beta_\psi \succcurlyeq_{par} \alpha_\psi$.

Let $\mathcal{B}' = \mathcal{B} \cup \{\alpha_\chi, \beta_\chi, \alpha_\theta, \beta_\theta\}$, where $\alpha_\chi \doteq \frac{1}{f_k}\alpha_\rho$ and $\beta_\chi \doteq \frac{1}{f_k}\beta_\rho$, and $\alpha_\theta \doteq \alpha_\chi + \beta_\varphi$ and $\beta_\theta \doteq \beta_\chi + \alpha_\varphi$, and χ, θ (which are thus in $\mathcal{L}^{\mathcal{B}'}$) are such that $\chi \doteq \frac{1}{f_k}\rho$ and $\theta \doteq \chi + \neg\varphi$, i.e., $\theta \doteq \chi + \varphi_k$. We have $f_k\theta \doteq f_k\chi + f_k\varphi_k \doteq \rho + f_k\varphi_k$ and thus $\psi \doteq f_k\theta$. This implies that θ is a strict statement and that $\beta_\theta \succcurlyeq_{par} \alpha_\theta$. Now, $\Gamma \vdash_{\mathcal{B}} \rho$ implies that $\Gamma \vdash_{\mathcal{B}'} \rho$ (because $\mathcal{B}' \subseteq \mathcal{B}$). Since $\chi \doteq \frac{1}{f_k}\rho$, we have $\Gamma \vdash_{\mathcal{B}'} \chi$, using the Point wise Multiplication inference rule, completing the proof. □

These lemmas lead to the completeness theorems.

Theorem 6.2: Completeness of Proof Theory (1)

Consider any $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ and any $\varphi \in \mathcal{L}^{\mathcal{A}}$. Then there exists $\mathcal{B} \supseteq \mathcal{A}$, with $\mathcal{B} \setminus \mathcal{A}$ finite such that $\Gamma \models \varphi \iff \Gamma \vdash_{\mathcal{B}} \varphi$.

Proof. $\Leftarrow$ follows by Proposition 6.33. To prove the converse, let us assume that $\Gamma \models \varphi$; we will show that $\mathcal{A}$ can be extended to $\mathcal{B}$ such that $\Gamma \vdash_{\mathcal{B}} \varphi$.

First let us consider the case when Γ is $\mathcal{H}(1)$-inconsistent. By Lemma 6.22 there exists $C' \subseteq \mathcal{V}$ and a $\mathcal{H}(1)$-inconsistent subset $\{\varphi_1, \ldots, \varphi_k\}$ of Γ, such that $(\{\varphi_1, \ldots, \varphi_k\}, C')$ is an inconsistency base for $(\Gamma, \mathcal{V})$. By Lemma 6.34, there exist

strictly positive functions $f_1, \ldots, f_k \in F$, set of alternatives $\mathcal{B} \supseteq \mathcal{A}$ with $\mathcal{B} \setminus \mathcal{A}$ finite, and strict preference statement ψ in $\mathcal{B}$ such that $\psi \doteq f_1\varphi_1 + \cdots + f_k\varphi_k$, and $\Gamma \vdash_{\mathcal{B}} \psi$ and $\beta_\psi \succcurlyeq_{par} \alpha_\psi$. Consider any $\gamma \in \mathcal{A}$. Then $\beta_\psi \succcurlyeq_{par} \alpha_\psi$ implies for all $c \in \mathcal{V}$, $\beta_\psi(c) - \alpha_\psi(c) \geq 0 = \gamma(c) - \gamma(c)$. The Pareto Difference inference rule then implies that $\Gamma \vdash_{\mathcal{B}} \gamma > \gamma$, since ψ is strict, and hence, by the Inconsistent Statement inference rule, $\Gamma \vdash_{\mathcal{B}} \varphi$, as required.

Now we consider the case when Γ is $\mathcal{H}(1)$-consistent. By Lemma 6.35, we have that there exists set of alternatives $\mathcal{B} \supseteq \mathcal{A}$ with $\mathcal{B} \setminus \mathcal{A}$ finite, and $\chi, \theta \in \mathcal{L}^{\mathcal{B}}$ such that $\Gamma \vdash_{\mathcal{B}} \chi$, and θ is strict, $\theta \doteq \chi + \neg\varphi$, and $\beta_\theta \succcurlyeq_{par} \alpha_\theta$. Then, by definition of $\neg\varphi$, we have $\alpha_\theta \doteq \alpha_\chi + \beta_\varphi$ and $\beta_\theta \doteq \beta_\chi + \alpha_\varphi$. This implies that for all $c \in \mathcal{V}$, $\beta_\chi(c) + \alpha_\varphi(c) \geq \alpha_\chi(c) + \beta_\varphi(c)$, and thus, for all $c \in \mathcal{V}$, $\beta_\chi(c) - \alpha_\chi(c) \geq \beta_\varphi(c) - \alpha_\varphi(c)$. Now, since $\theta \doteq \chi + \neg\varphi$ and θ is strict, if χ is non-strict then $\neg\varphi$ must be strict and so φ is non-strict. The Pareto Difference inference rule then implies that $\Gamma \vdash_{\mathcal{B}} \varphi$. If, on the other hand, χ is strict then the Pareto Difference inference rule implies that $\Gamma \vdash_{\mathcal{B}} \alpha_\varphi > \beta_\varphi$, and thus $\Gamma \vdash_{\mathcal{B}} \alpha_\varphi \geq \beta_\varphi$, using the From Strict to Non-Strict rule. Therefore, $\Gamma \vdash_{\mathcal{B}} \varphi$ whether φ is strict or non-strict. □

Let $\mathcal{A}^*$ be a set of alternatives including for each function $g : \mathcal{V} \to \mathbb{Q}^+$, an alternative α with, for all $c \in \mathcal{V}$, $\alpha(c) = g(c)$, and let $\mathcal{A}' = \mathcal{A} \cup \mathcal{A}^*$. Consider any $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ and any $\varphi \in \mathcal{L}^{\mathcal{A}}$. Then $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}^{\mathcal{A}'}$. If we use $\mathcal{A}'$ instead of $\mathcal{A}$ in the proofs of Lemma 6.34 and 6.35, and Theorem 6.2, we can use $\mathcal{B} = \mathcal{A}'$ in each case. This leads, for arbitrary Γ and φ, to: $\Gamma \models_{\mathcal{A}'} \varphi \iff \Gamma \vdash_{\mathcal{A}'} \varphi$, which since $\Gamma \models_{\mathcal{A}'} \varphi$ holds if and only if $\Gamma \models_{\mathcal{A}} \varphi$ holds, gives the following version of the completeness result.

Theorem 6.3: Completeness of Proof Theory (2)

For any $\mathcal{A}$, there exists $\mathcal{A}' \supseteq \mathcal{A}$ such that for any $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ and any $\varphi \in \mathcal{L}^{\mathcal{A}}$, $\Gamma \models \varphi \iff \Gamma \vdash_{\mathcal{A}'} \varphi$.

6.6 Discussion

Throughout this chapter, we considered lexicographic models $\mathcal{H}(1)$ in connection with strict and non-strict preference statements $\mathcal{L}^{\mathcal{A}}$. Here, because of the strong resemblance to CVO lexicographic models $\mathscr{L}$, we can observe that many

results previously proven in Section 4.3.2 for cvo lexicographic models $\mathscr{L}$ also hold true for models $\mathcal{H}(1)$. The cvo lexicographic models $\mathscr{L}$ are more general in the sense that value orders on the variable domains are not fixed but arbitrary total orders and part of the model. A similar composition operator and variable mapping as for models $\mathscr{L}$ is defined for $\mathcal{H}(1)$ models.

As a main result of this Chapter, the statements $\mathcal{L}^{\mathcal{A}}$ are strongly compositional and the algorithm of Section 4.2.4 can be applied to solve the Consistency and the Deduction Problem (which are mutually expressive for the considered case).

We describe in detail how to choose minimal extensions in the algorithm, and how to do ($\models^*$-) satisfaction checks. Here, we make use of inconsistency bases and Γ-allowed sequences. Interestingly, analysing these structures, we can observe that even for a set of inconsistent preference statements, the preference model created by the algorithm is the closest approximation we can have to a satisfying lexicographic preference model. Furthermore, we show that a preference language that gives order constraints on variables instead of alternatives, is equally expressive as the language $\mathcal{L}^{\mathcal{A}}$ for models $\mathcal{H}(1)$.

Strong consistency considers the existence of a preference model that involves all variables and satisfies the given preference statements. Naturally, we can apply the same algorithm as described earlier. Also, if a set of preference statements is strongly consistent, then the set of models of the preference statements that include all variables is the same as the set of maximal models of the preference statements. Thus, the consequences of the inference of either model set are the same. We also identified cases in which the inference based on models including all variables is the same as the inference with models of all sizes. These observations rely on properties deduced from inconsistency bases and Γ-allowed sequences.

We can thus conclude that inconsistency bases are a very helpful concept in understanding the structure of the Consistency Problem for statements $\mathcal{L}^{\mathcal{A}}$ and models $\mathcal{H}(1)$. It might be interesting to investigate this concept for cvo lexicographic models $\mathscr{L}$. Here, inconsistency bases would have to respect order constraints on value orders of variable domains. It is not obvious, how the concept of inconsistency bases can be transferred to the more general hierarchical models $\mathcal{H}(t)$ for $t > 1$. However, for the case of t-bound Pareto models $\mathcal{P}(t)$ with $t \geq 1$, identifying variables / variable sets, which cannot be included in any model satisfying the input preferences, is possible and discussed in Chapter 5.

The completeness results of the prove theory discussed in Section 6.5 show that there exists an equivalent syntactical definition of the inference discussed in this chapter, which so far has only been defined semantically.

Note that we only concentrated on statements in $\mathcal{L}^{\mathcal{A}}$ and did not consider the languages $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ for models $\mathcal{H}(1)$. These languages will be discussed in connection with models $\mathscr{L}$ in Chapter 8. However, because models $\mathscr{L}$ are a generalisation of models $\mathcal{H}(1)$ of sorts, we believe that many results developed for models $\mathscr{L}$ in Chapter 8 for languages $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ can also hold for models $\mathcal{H}(1)$.

Chapter 7

Hierarchical Model

In this chapter, we consider the Deduction and Consistency Problem for hierarchical models $\mathcal{H}(t)$ with $t > 1$ for input preference statements $\mathcal{L}^{\mathcal{A}}$ that are strict and non-strict comparisons on complete alternatives.

After a detailed introduction of deduction and consistency for hierarchical models $\mathcal{H}(t)$, we show that the Deduction Problem is coNP-complete, even if one restricts the cardinality of the equal-importance sets of variables to have at most two elements (Section 7.2). Recall from Chapter 6 that it is polynomial in many cases in which it is assumed that the user's ordering of variables is a total ordering, i.e., a fvo lexicographic models $\mathcal{H}(1)$. At the end of Section 7.1, we briefly mention the special case where a fixed equivalence relation on variables is given that specifies the possible level sets. In this case the problem is polynomial and the algorithm from Section 6.3.3 can be applied to solve it.

In Section 7.3, we focus on finding efficient algorithm approaches for the Consistency Problem (and thus the Deduction Problem). Here, we first describe a Mixed Integer Linear Program formulation and then approach the problem with a recursive search. The recursive search relies on a pruning of the search space that is based on specific properties of hierarchical models. Furthermore, a variant of the recursive search is described in which conflicting sets of variables are maintained, by which the search space can be pruned further. We then describe an experimental set up and runtime comparison of the developed approaches in Section 7.4. The last section concludes.

This chapter takes results and descriptions from [GWO16, WGO15] and [WG17].

7.1 Preliminaries

We consider preference models, based on an importance ordering of variables that is basically lexicographic, but involving a combination of variables which are at the same level in the importance ordering. In the papers that this chapter is based on, we called these “HCLP models", because models of a similar kind are considered in Hierarchical Constraint Logic Programming systems [WB93] (though we have abstracted away some details from the latter system).

Definition 7.1: Hierarchical Structures

Define a hierarchical structure to be a tuple $\mathcal{S} = \langle \mathcal{A}, \oplus, \mathcal{V} \rangle$. Here, $\mathcal{A}$ (the set of *alternatives*) is a (possibly infinite) set. $\mathcal{V}$ is a finite set of variables with the same domains and a fixed order $\geq$ on the variable's domain D. Elements in $\mathcal{A}$ are vectors over the common domain D of variables $X \in \mathcal{V}$, i.e., $\mathcal{A} \subseteq D^{|\mathcal{V}|}$. $\oplus$ is an associative, commutative and monotonic operation ($x \oplus y \geq z \oplus y$ if $x \geq z$) on the variable's domain D. Furthermore, we assume that D contains an identity element $0 \in D$ and at least one other element which we call "1" such that $1 > 0$.

Note that in practice this means that variables are commensurable such that an operator $\oplus$ exists for which combining variable values is reasonable. This can include, for example, variables with cost values, but also equal ordinal scales (e.g., small > medium > large, or good > medium > bad).

In this chapter, we assume that operation $\oplus$ can be computed in linear time (which holds for natural definitions of $\oplus$, including addition and max on $\mathbb{Q}^+$). The variables in $\mathcal{V}$ may be considered as representing criteria or objectives under which the alternatives are evaluated.

Example 7.1

Suppose, a user wants to buy a new prepay mobile phone SIM card and considers different providers based on the price per 10MB data usage (d), the price per text message (m) and the price per minute for calls to the same provider (c). These prices of d, m and c can be combined by addition. For any of the four price categories, the lower the price is the better it is for the customer. Consider four different options (providers) α, β, γ and δ

with the following prices in cent.

	α	β	γ	δ
d	18	15	13	14
m	15	17	15	13
c	10	11	14	15

In this context, the hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$ is given by the set of alternatives $\mathcal{A} = \{\alpha, \beta, \gamma, \delta\}$, the operator $\oplus$ being the ordinary addition on the integers and the set of variables $\mathcal{V} = \{d, m, c\}$ specifying the alternatives as in the table above.

For each subset C of $\mathcal{V}$ we define ordering $\succcurlyeq_C^{\oplus}$ on $\mathcal{A}$ by $\alpha \succcurlyeq_C^{\oplus} \beta$ if and only if $\bigoplus_{c \in C} \alpha(c) \geq \bigoplus_{c \in C} \beta(c)$. Relation $\succcurlyeq_C^{\oplus}$ represents how well the alternatives satisfy the set of variables C if the latter are considered equally important. $\succcurlyeq_C^{\oplus}$ is a total pre-order (a weak order, i.e., a transitive and complete binary relation). We write $\equiv_C^{\oplus}$ for the associated equivalence relation on $\mathcal{A}$, given by $\alpha \equiv_C^{\oplus} \beta \iff \alpha \succcurlyeq_C^{\oplus} \beta$ and $\beta \succcurlyeq_C^{\oplus} \alpha$. We write $\succ_C^{\oplus}$ for the associated strict weak ordering, defined by $\alpha \succ_C^{\oplus} \beta \iff \alpha \succcurlyeq_C^{\oplus} \beta$ and $\beta \not\succcurlyeq_C^{\oplus} \alpha$. Thus, $\alpha \equiv_C^{\oplus} \beta$ if and only if $\bigoplus_{c \in C} \alpha(c) = \bigoplus_{c \in C} \beta(c)$; and $\alpha \succ_C^{\oplus} \beta$ if and only if $\bigoplus_{c \in C} \alpha(c) > \bigoplus_{c \in C} \beta(c)$.

Recall from Definition 3.16 that a hierarchical model H based on $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$ is defined to be an ordered partition $(C_1, \ldots, C_k)$ of a subset of $\mathcal{V}$; we label this subset as $\sigma(H)$, so that $\sigma(H) = C_1 \cup \cdots \cup C_k$. Note that $\sigma(H)$ can be empty, which corresponds to the empty model $H = ()$. However, the variable sets within a model are non-empty. The sets C_i are called *level sets* or *levels of* H, which are thus non-empty, disjoint and have union $\sigma(H)$. If $c \in C_i$ and $c' \in C_j$, and $i < j$, then we say that c *appears before* c' (and c' *appears after* c) in H.

Recall from Definition 3.18 that the set of t-bound hierarchical models for $t \in \mathbb{N}$ is defined as the set of hierarchical models with level sets of maximum cardinality t, i.e., $\mathcal{H}(t) = \{H = (C_1, \ldots, C_l) \mid H \text{ is a hierarchical model and } |C_i| \leq t \; \forall i = 1, \ldots, l\}$. An element of $\mathcal{H}(1)$ thus corresponds to a sequence of singleton sets of variables; this special case has been discussed in the previous chapter. Note that, these models do not depend on $\oplus$ (since there is no combination of variables involved), so we were able to drop any mention of $\oplus$ in the previous chapter.

Associated with a hierarchical model $H = (C_1, \ldots, C_k)$ is an order relation $\succcurlyeq_H^{\oplus}$ on $\mathcal{A}$ as described in Definition 3.17:

$\alpha \succcurlyeq_H^{\oplus} \beta$ if and only if either:

(I) for all $i = 1, \ldots, k$, $\alpha \equiv_{C_i}^{\oplus} \beta$; or

(II) there exists some $i \in \{1, \ldots, k\}$ such that (i) $\alpha \succ_{C_i}^{\oplus} \beta$ and (ii) for all j with $1 \leq j < i$, $\alpha \equiv_{C_j}^{\oplus} \beta$.

Relation $\succcurlyeq_H^{\oplus}$ is a kind of lexicographic order on $\mathcal{A}$, where the values of variables in the level set C_i are first combined into a single value. $\succcurlyeq_H^{\oplus}$ is a total preorder on $\mathcal{A}$. We write $\equiv_H^{\oplus}$ for the associated equivalence relation (corresponding with condition (I)), and $\succ_H^{\oplus}$ for the associated strict weak order (corresponding with condition (II)), so that $\succcurlyeq_H^{\oplus}$ is the disjoint union of $\succ_H^{\oplus}$ and $\equiv_H^{\oplus}$. If $\sigma(H) = \emptyset$, i.e., $H = ()$ the empty model, then the first condition for $\alpha \succcurlyeq_H^{\oplus} \beta$ holds vacuously (since $k = 0$), so we have $\alpha \succcurlyeq_{()}^{\oplus} \beta$ for all $\alpha, \beta \in \mathcal{A}$, and $\succ_{()}^{\oplus}$ is the empty relation.

Preference Language Inputs: Consider the preference language $\mathcal{L}^{\mathcal{A}}$ of strict and non-strict comparisons over alternatives in $\mathcal{A}$ as defined in Section 3.2. We define $\mathcal{L}_{\geq}^{\mathcal{A}}$ to be the set of non-strict statements of the form $\alpha \geq \beta$ ("α is preferred to β"), for $\alpha, \beta \in \mathcal{A}$; we write $\mathcal{L}_{>}^{\mathcal{A}}$ for the set of strict statements of the form $\alpha > \beta$ ("α is strictly preferred to β"), for $\alpha, \beta \in \mathcal{A}$. Hence, $\mathcal{L}^{\mathcal{A}} = \mathcal{L}_{\geq}^{\mathcal{A}} \cup \mathcal{L}_{>}^{\mathcal{A}}$. Recall from Section 2.4.2 that since hierarchical models induce a total preorder on the alternatives $\mathcal{A}$, if φ is the preference statement $\alpha \geq \beta$, then $\neg\varphi$ is the preference statement $\beta > \alpha$. If φ is the preference statement $\alpha > \beta$, then $\neg\varphi$ is the preference statement $\beta \geq \alpha$. In the following, we sometimes write a preference statement $\varphi \in \mathcal{L}_{\geq}^{\mathcal{A}}$ as $\alpha_\varphi \geq \beta_\varphi$, and $\varphi \in \mathcal{L}_{>}^{\mathcal{A}}$ as $\alpha_\varphi > \beta_\varphi$ for $\alpha_\varphi, \beta_\varphi \in \mathcal{A}$. We denote the non-strict version of preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ by $\Gamma^{(\geq)}$, i.e., $\Gamma^{(\geq)} = \{\alpha_\varphi \geq \beta_\varphi \mid \varphi \in \Gamma\}$ (see Definition 3.4).

Satisfaction of preference statements: For a hierarchical model H over the hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$, we say that H *satisfies* $\alpha \geq \beta$ (written $H \models^{\oplus} \alpha \geq \beta$), if $\alpha \succcurlyeq_H^{\oplus} \beta$ holds. Similarly, we say that H *satisfies* $\alpha > \beta$ (written $H \models^{\oplus} \alpha > \beta$), if $\alpha \succ_H^{\oplus} \beta$. For $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, we say that H *satisfies* Γ (written $H \models^{\oplus} \Gamma$), if H satisfies φ for all $\varphi \in \Gamma$. If $H \models^{\oplus} \varphi$, then we sometimes say that H *is a model of* φ (and similarly, if $H \models^{\oplus} \Gamma$).

Example 7.2

Consider Example 7.1 of a user choosing between different providers to buy a prepay SIM card.

Suppose, the user is not interested in using data and has as many call minutes as text messages, i.e., the prices m and c are equally important. She can express her preferences by the corresponding hierarchical model $H = (\{m, c\})$ in $\mathcal{H}(t)$ with $t \geq 2$. Since $\alpha(m) + \alpha(c) = 25 < \beta(m) + \beta(c) = 28 = \delta(m) + \delta(c) = 28 < \gamma(m) + \gamma(c) = 29$, H satisfies $\gamma \prec_H^{\oplus} \beta \equiv_H^{\oplus} \delta \prec_H^{\oplus} \alpha$. The variables involved in H are $\sigma(H) = \{m, c\}$.

If the user is most interested in the text message prices, and only if these are equal in the call prices, and only if these are also equal in the data prices, then the corresponding hierarchical model is $H' = (\{m\}, \{c\}, \{d\})$ in $\mathcal{H}(t)$ with $t \geq 1$. The induced order relation for this model satisfies $\beta \prec_{H'}^{\oplus} \gamma \prec_{H'}^{\oplus} \alpha \prec_{H'}^{\oplus} \delta$, since $\delta(m) < \alpha(m) = \gamma(m) < \beta(m)$ and $\alpha(c) < \gamma(c)$. The variables involved in H' are $\sigma(H') = \{d, m, c\}$.

Satisfaction of negated preference statements behaves as one would expect:

Lemma 7.1. *Let H be a hierarchical model over hierarchical structure $\mathcal{S}$. Then, H satisfies φ if and only if H does not satisfy $\neg\varphi$.*

Proof. Write $\mathcal{S}$ as $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$. First show that, for any $\alpha, \beta \in \mathcal{A}$, H satisfies $\alpha \geq \beta$ if and only if H does not satisfy $\beta > \alpha$. We have that H satisfies $\alpha \geq \beta$ if and only if $\alpha \succcurlyeq_H^{\oplus} \beta$, which, since $\succcurlyeq_H^{\oplus}$ is a weak order, is if and only if $\beta \not\succcurlyeq_H^{\oplus} \alpha$, i.e., H does not satisfy $\beta > \alpha$. It immediately follows that H satisfies $\alpha > \beta$ if and only if H does not satisfy $\beta \geq \alpha$. □

As for other model types, for preference statements Γ and statement φ we say that $\Gamma \models_{\mathcal{H}(t)}^{\oplus} \varphi$, if $H \models^{\oplus} \varphi$ for every $H \in \mathcal{H}(t)$. Also, Γ is $\mathcal{H}(t)$-inconsistent for operator $\oplus$, if there exists no $H \in \mathcal{H}(t)$ such that $H \models^{\oplus} \Gamma$. The next proposition shows the relation between deduction and consistency for hierarchical models based on statements $\mathcal{L}^{\mathcal{A}}$.

Theorem 7.1: Mutual Expressiveness of Consistency and Deduction

Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ be a set of preference statements and $\varphi \in \mathcal{L}^{\mathcal{A}}$. $\Gamma \models^{\oplus}_{\mathcal{H}(t)} \varphi$ if and only if $\Gamma \cup \{\neg\varphi\}$ is $\mathcal{H}(t)$-inconsistent for operator $\oplus$.

Proof. Suppose that $\Gamma \models^{\oplus}_{\mathcal{H}(t)} \varphi$. By definition, H satisfies φ for every $H \in \mathcal{H}(t)$ satisfying (every element of) Γ. Thus, using Lemma 7.1, there exists no $H \in \mathcal{H}(t)$ that satisfies Γ and $\neg\varphi$, which implies that $\Gamma \cup \{\neg\varphi\}$ is $\mathcal{H}(t)$-inconsistent for operator $\oplus$.

Conversely, suppose $\Gamma \cup \{\neg\varphi\}$ is $\mathcal{H}(t)$-inconsistent for operator $\oplus$. By definition, there exists no $H \in \mathcal{H}(t)$ that satisfies $\Gamma \cup \{\neg\varphi\}$. Thus, every $H \in \mathcal{H}(t)$ that satisfies Γ does not satisfy $\neg\varphi$, and therefore satisfies φ, by Lemma 7.1. Hence, $\Gamma \models^{\oplus}_{\mathcal{H}(t)} \varphi$. □

We formulate the Preference Consistency and Deduction (decision) Problems for classes $\mathcal{H}(t)$ as follows.

$\mathcal{H}(t)$ **Preference Consistency Problem ($\mathcal{H}(t)$-PCP):** Given a hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$, a constant $t \in \{1, \ldots, |\mathcal{V}|\}$ and a set of preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$. Is Γ $\mathcal{H}(t)$-consistent for operator $\oplus$?

$\mathcal{H}(t)$ **Preference Deduction Problem ($\mathcal{H}(t)$-PDP):** Given a hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$, a constant $t \in \{1, \ldots, |\mathcal{V}|\}$, some preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$ and $\varphi \in \mathcal{L}^{\mathcal{A}} \setminus \Gamma$. Does $\Gamma \models^{\oplus}_{\mathcal{H}(t)} \varphi$? In other words, does $H \models^{\oplus} \varphi$ hold for all $H \in \mathcal{H}(t)$ with $H \models^{\oplus} \Gamma$?

Note that, the empty model $H = ()$ always satisfies non-strict statements, but never satisfies strict statements. Thus, $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}_{\geq}$ is always consistent. It is easy to see that Γ is $\mathcal{H}(t)$-consistent, if Γ is $\mathcal{H}(s)$-consistent for some $s < t$. Here, the class of hierarchical models $\mathcal{H}(1)$ consists of fvo lexicographic models that imply the usual lexicographic order relations. Thus, if preference statements Γ are consistent with respect to fvo lexicographic models $\mathcal{H}(1)$ (see Definition 3.14), then Γ is consistent with respect to hierarchical models in $\mathcal{H}(t)$.

Example 7.3

Consider the hierarchical structure of Example 7.1.

Suppose, the user states that she prefers α to β, i.e. $\alpha \geq \beta$, and strictly prefers β to γ, i.e. $\beta > \gamma$. Only the hierarchical models $(\{c\}, \dots)$, $(\{m\}, \dots)$, $(\{c, m\}, \dots)$ or $(\{d, m, c\})$ satisfy $\alpha \geq \beta$, where "..." signifies any possible extension. Only the hierarchical model $(\{c\}, \dots)$, $(\{c, d\}, \dots)$ or $(\{c, m\}, \dots)$ satisfy $\beta > \gamma$. Thus, the models $(\{c\}, \dots)$ and $(\{c, m\}, \dots)$ are the only ones that satisfy the set $\Gamma = \{\alpha \geq \beta, \beta > \gamma\}$ of the user's preferences.

Let $t \in \{1, 2, 3\}$. Then $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \delta \geq \beta$ since the model $H = (\{c\}) \in \mathcal{H}(1) \subseteq \mathcal{H}(t)$ satisfies Γ and $\beta \succ^{\oplus}_{H} \delta$, i.e., $H \models^{\oplus} \beta > \delta$. Furthermore, $\Gamma \not\models^{\oplus}_{\mathcal{H}(2)} \beta \geq \delta$ since the model $H' = (\{c, m\}, \{d\}) \in \mathcal{H}(2)$ satisfies Γ and $\delta \succ^{\oplus}_{H'} \beta$, i.e.,$H' \models^{\oplus} \delta > \beta$. However, we can infer $\Gamma \models^{\oplus}_{\mathcal{H}(1)} \beta \geq \delta$, and even $\Gamma \models^{\oplus}_{\mathcal{H}(1)} \beta > \delta$, since all Γ-satisfying hierarchical model in $\mathcal{H}(1)$, i.e., $(\{c\})$, $(\{c\}, \{m\})$, $(\{c\}, \{d\})$, $(\{c\}, \{m\}, \{d\})$, and $(\{c\}, \{d\}, \{m\})$, satisfy the relation $\beta > \delta$.

Fixed Equivalence Classes of Variables Let $\equiv$ be an equivalence relation on $\mathcal{V}$. We define $\mathcal{H}(\equiv)$ to be the set of all hierarchical models $(C_1, \dots, C_k)$ such that each C_i is an equivalence class with respect to $\equiv$. It is easy to see that the relation $\models^{\oplus}_{\mathcal{H}(\equiv)}$ is the same as the relation $\models^{\oplus}_{\mathcal{H}'(1)}$ where $\mathcal{H}'(1)$ is defined as follows. $\mathcal{V}'$ is in 1-1 correspondence with the set of $\equiv$-equivalence classes of $\mathcal{V}$. If E is the $\equiv$-equivalence class of $\mathcal{V}$ corresponding with $e \in \mathcal{V}'$ then, for $\alpha \in \mathcal{A}$, $\alpha(e)$ is defined to be $\bigoplus_{c \in E} \alpha(c)$. $\mathcal{H}'(1)$ is the set of singleton hierarchical models on variables $\mathcal{V}'$.

In Chapter 6, we showed that the Consistency Problem (and thus also the Deduction Problem) is polynomial for $\models_{\mathcal{H}(1)}$. Thus it is polynomial also for $\models^{\oplus}_{\mathcal{H}(\equiv)}$, for any equivalence relation $\equiv$ and the polynomial time algorithm from Section 6.3.3 can be applied. In contrast, it is coNP-complete for $\models$ being $\models^{\oplus}_{\mathcal{H}(t)}$ when $t > 1$, as we show below in the next section.

7.2 coNP-completeness of $\mathcal{H}(t)$-Deduction for $t > 1$

In this section, we prove the following coNP-completeness result for Deduction for $\models^{\oplus}_{\mathcal{H}(t)}$ with $t > 1$ (as defined in Section 7.1) by a reduction from 3-SAT.

Theorem 7.2: coNP-completeness of PDP for Hierarchical Models

The $\mathcal{H}(t)$ Preference Deduction Problem is coNP-complete for any $t > 1$, even if we restrict the language to non-strict preference statements $\mathcal{L}^{\mathcal{A}}_{\geq}$.

We have that $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$ if and only if there exists a hierarchical model $H \in \mathcal{H}(t)$ such that $H \models^{\oplus} \Gamma$ and $H \not\models^{\oplus} \beta \geq \alpha$. By our assumption, the operator $\oplus$ is computable in linear time. To check whether a preference statement is satisfied by a hierarchical model, the $\oplus$-combinations of values for variables in the same level sets have to be computed and compared. Thus, for any given $H \in \mathcal{H}(t)$, checking that $H \models^{\oplus} \Gamma$ and $H \not\models^{\oplus} \beta \geq \alpha$ can be performed in polynomial time (in the number of variables and preference statements). This implies that determining if $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$ holds is in NP.

Given an arbitrary 3-SAT instance, we will show that we can construct a set Γ of non-strict statements and a statement $\beta \geq \alpha$ such that the 3-SAT instance has a satisfying truth assignment if and only if $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$ (see Proposition 7.5 below). This then implies that determining if $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$ holds is NP-complete, and thus determining if $\Gamma \models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$ holds is coNP-complete.

The idea behind the reduction: Consider an arbitrary 3-SAT instance based on propositional variables $p_1, \ldots, p_r$ that consists of clauses Λ_j, for $j = 1, \ldots, s$. With each propositional variable p_i, we associate two variables q_i^+ and q_i^-, where q_i^- corresponds with literal $\neg p_i$, and q_i^+ corresponds with literal p_i. We construct a (polynomial size) set $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}_{\geq}$ of preference statements, which is the disjoint union of sets Γ_1, Γ_2 and Γ_3, and we construct a non-strict statement $\beta \geq \alpha$. For the remainder of this section, let H be an arbitrary hierarchical model in $\mathcal{H}(t)$. Γ_1 is chosen so that if $H \models^{\oplus} \Gamma_1$ then, for each $i = 1, \ldots, r$, $\sigma(H)$ cannot contain both q_i^+ and q_i^-, i.e., q_i^+ and q_i^- do not both appear in H. (Recall H is an ordered partition of $\sigma(H)$, so that $\sigma(H)$ is the set of variables that appear in H.) We choose Γ_2 such that, if $H \models^{\oplus} \Gamma_2$ and $H \models^{\oplus} \alpha > \beta$, then $\sigma(H)$ contains either q_i^+ or q_i^-. Together, this implies that, if $H \models^{\oplus} \Gamma$ and $H \not\models^{\oplus} \beta \geq \alpha$ (i.e.,

$H \models^{\oplus} \alpha > \beta$), then for each propositional variable p_i, model H involves either q_i^+ or q_i^-, but not both. Γ_3 is used to make the correspondence with the clauses. For instance, if one of the clauses is $p_2 \vee \neg p_5 \vee p_6$, then any hierarchical model $H \in \mathcal{H}(t)$ of $\Gamma \cup \{\alpha > \beta\}$ will involve either q_2^+, q_5^-, or q_6^+.

Suppose that H satisfies Γ but not $\beta \geq \alpha$. We can generate a satisfying assignment of the 3-SAT instance, by assigning p_i to 1 (TRUE) if and only if q_i^+ appears in H.

The monotonicity assumption for operation $\oplus$ implies that $1 \oplus 1 > 0$, since we have $1 \oplus 1 \geq 1 \oplus 0 = 1 > 0$. In fact, in the proof below we do not need to assume monotonicity of $\oplus$; it is sufficient to just assume that $1 \oplus 1 > 0$.

We describe the construction more formally in the following.

Defining $\mathcal{A}$ and $\mathcal{V}$: The set of alternatives $\mathcal{A}$ is defined to be the union of the following sets, where each of these alternatives is defined below.

- $\{\alpha, \beta\} \cup \{\alpha_i, \beta_i, \delta_i \mid i = 1, \ldots, r\}$,
- $\{\gamma_i^k \mid i = 1, \ldots, r, k = 1, \ldots, t-1\}$,
- $\{\theta_j, \tau_j \mid j = 1, \ldots, s\}$.

We define the set of variables $\mathcal{V}$ to be $\{c^*\} \cup \{q_i^+, q_i^- \mid i = 1, \ldots, r\} \cup A_1 \cup \cdots \cup A_r$, where $A_i = \{a_i^k \mid k = 1, \ldots, t-1\}$.

Both $\mathcal{A}$ and $\mathcal{V}$ are of polynomial size.

Satisfying $\alpha > \beta$: The values of α and β on the variables are defined as follows.

- $\alpha(c^*) = 1$, and for all $c \in \mathcal{V} - \{c^*\}$, $\alpha(c) = 0$.
- For all $c \in \mathcal{V}$, $\beta(c) = 0$.

It immediately follows that: $H \models^{\oplus} \alpha > \beta \iff \sigma(H) \ni c^*$, for $H \in \mathcal{H}(t)$.

The construction of Γ_1: We define $\Gamma_1 = \bigcup_{i=1}^{r} \Gamma_1^i$ where, for each $i = 1, \ldots, r$, we define $\Gamma_1^i = \{\delta_i \geq \gamma_i^k,\ \gamma_i^k \geq \delta_i \mid k = 1, \ldots, t-1\}$. We make use of auxiliary variables $A_i = \{a_i^1, \ldots, a_i^{t-1}\}$. The values of γ_i^k and δ_i on the variables are defined as follows:

- $\gamma_i^k(a_i^k) = 1$, and for all $c \in \mathcal{V} - \{a_i^k\}$ we set $\gamma_i^k(c) = 0$.
- $\delta_i(q_i^+) = \delta_i(q_i^-) = 1$, and for other $c \in \mathcal{V}$, $\delta_i(c) = 0$.

Thus, for any $B \subseteq A_i$, we have $\bigoplus_{a\in B} \delta_i(a) \oplus \delta_i(q_i^+) = 0 \oplus \cdots \oplus 0 \oplus 1 = 1$. Similarly, $\bigoplus_{a\in B} \delta_i(a) \oplus \delta_i(q_i^-) = 1$. Furthermore, $\bigoplus_{a\in B} \gamma_i^k(a) \oplus \gamma_i^k(q_i^+) = 1 \Leftrightarrow a_i^k \in B$ and $\bigoplus_{a\in B} \gamma_i^k(a) \oplus \gamma_i^k(q_i^-) = 1 \Leftrightarrow a_i^k \in B$. This helps us to prove the following lemma.

Lemma 7.2. *$H \models^{\oplus} \Gamma_1^i$ if and only if either (i) $\sigma(H)$ does not contain any element in A_i or q_i^+ or q_i^-, i.e., $\sigma(H) \cap (A_i \cup \{q_i^+, q_i^-\}) = \emptyset$; or (ii) $A_i \cup \{q_i^+\}$ is a level of H, and $\sigma(H) \not\supseteq q_i^-$; or (iii) $A_i \cup \{q_i^-\}$ is a level of H, and $\sigma(H) \not\supseteq q_i^+$. In particular, if $H \models^{\oplus} \Gamma_1^i$, then $\sigma(H)$ does not contain both q_i^+ and q_i^-.*

Proof. Consider any $H \in \mathcal{H}(t)$, so that for each level set E in H we have $|E| \leq t$. We have that $H \models^{\oplus} \Gamma_1^i$ if and only if for each level set E in H and for all $k = 1, \ldots, t-1$, $\delta_i \equiv_E^{\oplus} \gamma_i^k$. Now, $\delta_i \equiv_E^{\oplus} \gamma_i^k$ if and only if $\bigoplus_{c\in E} \delta_i(c) = \bigoplus_{c\in E} \gamma_i^k(c)$. Also, $\bigoplus_{c\in E} \delta_i(c) = 0$ if E contains neither q_i^+ nor q_i^-; and $\bigoplus_{c\in E} \delta_i(c) = 1 \oplus 1 > 0$ if E contains both q_i^+ and q_i^-; and $\bigoplus_{c\in E} \delta_i(c) = 1$ if E contains either q_i^+ or q_i^-, but not both. $\bigoplus_{c\in E} \gamma_i^k(c)$ equals 1 if and only if E contains a_i^k, and equals 0 otherwise.

This implies that, if for all $k = 1, \ldots, t-1$, $\delta_i \equiv_E^{\oplus} \gamma_i^k$ and E contains q_i^+ or q_i^-, then for all $k = 1, \ldots, t-1$, E contains a_i^k, and so $E \supseteq A_i$. Because of the condition that $|E| \leq t$ (since $H \in \mathcal{H}(t)$), and $|A_i| = t - 1$, we then have that E equals either $A_i \cup \{q_i^+\}$ or $A_i \cup \{q_i^-\}$.

Similarly, if for all $k = 1, \ldots, t-1$, $\delta_i \equiv_E^{\oplus} \gamma_i^k$ and E contains a_i^k for some $k \in \{1, \ldots, t-1\}$, then E contains q_i^+ or q_i^-, and so, by the previous paragraph, E equals either $A_i \cup \{q_i^+\}$ or $A_i \cup \{q_i^-\}$.

Thus, if $H \models^{\oplus} \Gamma_1^i$, then for at most one level E of H do we have $E \cap (A_i \cup \{q_i^+, q_i^-\})$ non-empty (else we would have two levels both containing A_i, contradicting disjointness of levels); also if $E \cap (A_i \cup \{q_i^+, q_i^-\})$ is non-empty then E equals either $A_i \cup \{q_i^+\}$ or $A_i \cup \{q_i^-\}$. In particular, if $H \models^{\oplus} \Gamma_1^i$, then $\sigma(H)$ does not contain both q_i^+ and q_i^-.

Regarding the converse, let us suppose first that (i) $\sigma(H)$ does not intersect with $A_i \cup \{q_i^+, q_i^-\}$. Then for all levels E of H, and for all $k = 1, \ldots, t-1$, we have $\bigoplus_{c\in E} \delta_i(c) = \bigoplus_{c\in E} \gamma_i^k(c) = 0$, and thus $\delta_i \equiv_E^{\oplus} \gamma_i^k$, which implies $H \models^{\oplus} \Gamma_1^i$.

Now suppose (ii) that $A_i \cup \{q_i^+\}$ is a level E' of H and $\sigma(H) \not\supseteq q_i^-$. Then every

other level E is disjoint from $A_i \cup \{q_i^+, q_i^-\}$, so for all $k = 1, \ldots, t-1$, $\bigoplus_{c \in E} \delta_i(c) = \bigoplus_{c \in E} \gamma_i^k(c) = 0$, and thus $\delta_i \equiv_E^{\oplus} \gamma_i^k$. Also, $\bigoplus_{c \in E'} \delta_i(c) = \bigoplus_{c \in E'} \gamma_i^k(c) = 1$, and thus $H \models^{\oplus} \Gamma_1^i$. Case (iii), when $A_i \cup \{q_i^-\}$ is a level E' of H and $\sigma(H) \not\ni q_i^+$, is essentially identical to Case (ii), just switching the roles of q_i^+ and q_i^+. We have proven that $H \models^{\oplus} \Gamma_1^i$, if (i), (ii) or (iii) hold. □

The construction of Γ_2: For each $i = 1, \ldots, r$, define φ_i to be $\beta_i \geq \alpha_i$. We let $\Gamma_2 = \{\varphi_i \mid i = 1, \ldots, r\}$. The values of α_i and β_i on the variables are defined as follows.

- $\alpha_i(c^*) = 1$, and for all $c \in \mathcal{V} - \{c^*\}$, $\alpha_i(c) = 0$.
- $\beta_i(q_i^+) = \beta_i(q_i^-) = 1$, and for all $c \in \mathcal{V} - \{q_i^+, q_i^-\}$, $\beta_i(c) = 0$.

Thus, similarly to the previous observations for Γ_1, $\beta_i(c^*) \oplus \beta_i(q_i^+) = \beta_i(c^*) \oplus \beta_i(q_i^-) = 1$ and $\alpha_i(c^*) \oplus \alpha_i(q_i^+) = \alpha_i(c^*) \oplus \alpha_i(q_i^-) = 1$. Also, $\alpha_i(q_i^+) \oplus \alpha_i(q_i^-) = 0$ and $\beta_i(q_i^+) \oplus \beta_i(q_i^-) \geq 1$, because of the monotonicity of $\oplus$, and $\alpha_i(c^*) \oplus \alpha_i(q_i^+) \oplus \alpha_i(q_i^-) = 1$ and $\beta_i(c^*) \oplus \beta_i(q_i^+) \oplus \beta_i(q_i^-) \geq 1$.

The following result easily follows.

Lemma 7.3. *If q_i^+ or q_i^- appears before c^* in H, then $H \models^{\oplus} \varphi_i$. If $\sigma(H) \ni c^*$ and $H \models^{\oplus} \varphi_i$, then $\sigma(H) \ni q_i^+$ or $\sigma(H) \ni q_i^-$.*

Proof. Consider any $H \in \mathcal{H}(t)$, and consider any $i \in \{1, \ldots, r\}$. Then the following hold for any level set E in H.

(I) If E does not contain any of $\{c^*, q_i^+, q_i^-\}$, then $\bigoplus_{c \in E} \alpha_i(c) = \bigoplus_{c \in E} \beta_i(c) = 0$, so $\alpha_i \equiv_E^{\oplus} \beta_i$.

(II) If E contains c^* but neither of q_i^+ or q_i^-, then $\bigoplus_{c \in E} \alpha_i(c) = 1$ and $\bigoplus_{c \in E} \beta_i(c) = 0$, so $\beta_i \not\succeq_E^{\oplus} \alpha_i$.

(III) If E contains q_i^+ or q_i^- but not c^*, then $\bigoplus_{c \in E} \alpha_i(c) = 0$ and $\bigoplus_{c \in E} \beta_i(c) > 0$ using the fact that $1 \oplus 1 > 0$, so $\beta_i \succ_E^{\oplus} \alpha_i$.

Assume that $\sigma(H) \ni c^*$. If $\sigma(H) \cap \{q_i^+, q_i^-\} = \emptyset$, then by considering the level containing c^* we can see, using (I) and (II), that $\beta_i \not\succeq_H^{\oplus} \alpha_i$, so $H \not\models^{\oplus} \varphi_i$. This proves the second part of the lemma.

If q_i^+ or q_i^- (or both) appear before c^* in H then (I) and (III) imply that $\beta_i \succ_H^{\oplus} \alpha_i$ and thus $H \models^{\oplus} \varphi_i$. This proves the first part of the lemma. □

The construction of Γ_3: We define a function Q over all literals by $Q(p_i) = q_i^+$ and $Q(\neg p_i) = q_i^-$, for each $i = 1, \ldots, r$. Let us write the jth clause as $l_1 \vee l_2 \vee l_3$ for literals l_1, l_2 and l_3. Define $Q_j = \{Q(l_1), Q(l_2), Q(l_3)\}$. For example, if the jth clause was $p_2 \vee \neg p_5 \vee p_6$ then $Q_j = \{q_2^+, q_5^-, q_6^+\}$. We define ψ_j to be the statement $\tau_j \geq \theta_j$, and $\Gamma_3 = \{\psi_j \mid j = 1, \ldots s\}$, where the values of θ_j and τ_j are given as follows.

- $\theta_j(c^*) = 1$, and $\theta_j(c) = 0$ for all $c \in \mathcal{V} - \{c^*\}$.
- $\tau_j(q) = 1$ for $q \in Q_j$, and $\tau_i(c) = 0$ for all $c \in \mathcal{V} - Q_j$.

Lemma 7.4. *If some element of Q_j appears in H before c^*, and no level of H contains more than one element of Q_j, then $H \models^{\oplus} \psi_j$. If $\sigma(H) \ni c^*$ and $H \models^{\oplus} \psi_j$ then $\sigma(H)$ contains some element of Q_j.*

Proof. The proof of this result is similar to that of Lemma 7.3. Consider any $H \in \mathcal{H}(t)$ and the jth clause Λ_j of the 3-SAT instance. Then the following hold for any level set E in H.

(I) If E does not contain any element of $Q_j \cup \{c^*\}$, then $\bigoplus_{c \in E} \theta_j(c) = \bigoplus_{c \in E} \tau_j(c) = 0$ so $\theta_j \equiv_E^{\oplus} \tau_j$.

(II) If E contains c^* but no element of Q_j, then $\bigoplus_{c \in E} \theta_j(c) = 1$ and $\bigoplus_{c \in E} \tau_j(c) = 0$, so $\tau_j \not\succ_E^{\oplus} \theta_j$.

(III) If E contains exactly one element of Q_j but not c^*, then $\bigoplus_{c \in E} \theta_j(c) = 0$ and $\bigoplus_{c \in E} \tau_j(c) = 1$, so $\tau_j \succ_E^{\oplus} \theta_j$.

Assume that $\sigma(H) \ni c^*$. If $\sigma(H) \cap Q_j = \emptyset$, then by considering the level containing c^* we can see, using (I) and (II), that $\tau_j \not\succ_E^{\oplus} \theta_j$, so $H \not\models^{\oplus} \psi_j$. This argument proves that if $\sigma(H) \ni c^*$ and $H \models^{\oplus} \psi_j$, then $\sigma(H)$ contains some element of Q_j.

If some element of Q_j appears in H before c^*, and no level of H contains more than one element of Q_j, then (I) and (III) imply that $\tau_j \succ_H^{\oplus} \theta_j$ and thus $H \models^{\oplus} \varphi_i$. This proves the first part of the lemma. □

We set $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$. The following result implies that the Deduction Problem is coNP-complete (even if we restrict to the case when $\Gamma \cup \{\varphi\} \subseteq \mathcal{L}_{\geq}^{\mathcal{A}}$).

Proposition 7.5. *Using the notation defined above, the 3-SAT instance is satisfiable if and only if $\Gamma \not\models_{\mathcal{H}(t)}^{\oplus} \beta \geq \alpha$.*

Proof. First let us assume that $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$. Then by definition, there exists a hierarchical model $H \in \mathcal{H}(t)$ with $H \models^{\oplus} \Gamma$ and $H \not\models^{\oplus} \beta \geq \alpha$. Since $H \not\models^{\oplus} \beta \geq \alpha \iff H \models^{\oplus} \alpha > \beta$ because hierarchical models imply a total order on the alternatives, we have $H \models^{\oplus} \Gamma \cup \{\alpha > \beta\}$. Because $H \models^{\oplus} \alpha > \beta$, we have $\sigma(H) \ni c^*$.

Because also $H \models^{\oplus} \Gamma^i_2$, either $\sigma(H) \ni q^+_i$ or $\sigma(H) \ni q^-_i$, by Lemma 7.3. Since $H \models^{\oplus} \Gamma^i_1$, the set $\sigma(H)$ does not contain both q^+_i and q^-_i, by Lemma 7.2.

Let us define a truth function $f : \mathcal{P} \to \{0, 1\}$, where $\mathcal{P}$ is the set of propositional variables, as follows: $f(p_i) = 1 \iff \sigma(H) \ni q^+_i$. Since $\sigma(H)$ contains exactly one of q^+_i and q^-_i, we have $f(p_i) = 0 \iff \sigma(H) \ni q^-_i$. We extend f to negative literals in the obvious way: $f(\neg p_i) = 1 - f(p_i)$, and thus, $f(\neg p_i) = 1 \iff \sigma(H) \ni q^-_i$.

Since $H \models^{\oplus} \Gamma_3$ and $\sigma(H) \ni c^*$, $\sigma(H)$ contains at least one element of each Q_j, by Lemma 7.4. Thus, $f(l) = 1$ for at least one literal l in every clause, and hence f satisfies all clauses Λ_j. We have shown that f satisfies each clause of the 3-SAT instance, proving that the instance is satisfiable.

Conversely, suppose that the 3-SAT instance is satisfiable, so there exists a truth function f satisfying it. We will construct a hierarchical model $H \in \mathcal{H}(t)$ such that $H \models^{\oplus} \Gamma \cup \{\beta > \alpha\}$, and thus $H \not\models^{\oplus} \alpha \geq \beta$, proving that $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \alpha \geq \beta$.

For $i = 1, \ldots, r$, let $S_i = A_i \cup \{q^+_i\}$ if $f(p_i) = 1$, and otherwise, let $S_i = A_i \cup \{q^-_i\}$. Thus, if $f(p_i) = 1$ then $Q(p_i) \in S_i$; and if $f(\neg p_i) = 1$ then $Q(\neg p_i) \in S_i$. We then define H to be the sequence $S_1, S_2, \ldots, S_r, \{c^*\}$. Since $\sigma(H) \ni c^*$, we have that $H \models^{\oplus} \alpha > \beta$. By Lemma 7.2, for all $i = 1, \ldots, r$, $H \models^{\oplus} \Gamma^i_1$ and so $H \models^{\oplus} \Gamma_1$. By Lemma 7.3, for all $i = 1, \ldots, r$, $H \models^{\oplus} \varphi_i$, so $H \models^{\oplus} \Gamma_2$.

Consider any $j \in \{1, \ldots, s\}$, and, as above, write the jth clause as $l_1 \vee l_2 \vee l_3$. Truth assignment f satisfies this clause, so there exists $k \in \{1, 2, 3\}$ such that $f(l_k) = 1$. Then $Q(l_k)$ appears in H before c^*, so, by Lemma 7.4, $H \models^{\oplus} \psi_j$. Thus $H \models^{\oplus} \Gamma_3$.

Since $\Gamma = \Gamma_1 \cup \Gamma_2 \cup \Gamma_3$, we have shown that $H \models^{\oplus} \Gamma \cup \{\alpha > \beta\}$, proving that $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \beta \geq \alpha$. □

Example 7.4

Let $(p_1 \vee p_2 \vee \neg p_3) \wedge (\neg p_1 \vee p_2 \vee p_3)$ be an instance of 3-SAT with the three

propositional variables p_1, p_2, p_3 and clauses Λ_1, Λ_2. From this we construct a $\mathcal{H}(2)$-Deduction instance as in the previous paragraphs. Corresponding to the two possible assignments of each of the propositional variables p_1, p_2, p_3, we construct variables q_1^+, q_2^+, q_3^+ and q_1^-, q_2^-, q_3^-. We also introduce the additional variables c^* and $A_1 = \{a_1^1, a_2^1, a_3^1\}$. Furthermore, we construct alternatives α, β, $\alpha_1, \alpha_2, \alpha_3$, $\beta_1, \beta_2, \beta_3$, $\delta_1, \delta_2, \delta_3$, $\gamma_1^1, \gamma_2^1, \gamma_3^1$, θ_1, θ_2, τ_1, τ_2 for the preference statements $\alpha > \beta$, Γ_1, Γ_2 and Γ_3 as follows:

			Γ_2						Γ_1						Γ_3			
	α	β	α_1	β_1	α_2	β_2	α_3	β_3	δ_1	γ_1^1	δ_2	γ_2^1	δ_3	γ_3^1	θ_1	τ_1	θ_2	τ_2
q_1^+	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	0
q_2^+	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	1
q_3^+	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1
q_1^-	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	1
q_2^-	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
q_3^-	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0
c^*	1	0	1	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0
a_1^1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
a_2^1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
a_3^1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Here, the values of τ_1 and τ_2 correspond to the occurrences of the literals p_i or $\neg p_i$ in the clauses Λ_1 and Λ_2, respectively. Since the statement $\alpha > \beta$ is strict, the variable c^* has to be included in any satisfying hierarchical model. For the satisfaction of all preference statements Γ_1, Γ_2 and Γ_3, the same principal applies. To satisfy a non-strict preference statement $\nu \leq \rho$ in Γ, the first level set that contains variables with value 1 on ν must also contain at least as many variables with value 1 on ρ. The satisfaction of preference statement $\alpha_1 \leq \beta_1$, e.g., enforces that the same level set containing c^* (where $\alpha_1(c^*) = 1$) must also contain at least one of the variables q_1^+ or q_1^- (where $\beta_1(q_1^+) = 1$ and $\beta_1(q_1^-) = 1$). The assignment $p_1 =$ *true*, $p_2 =$ *true*, $p_3 =$ *false* satisfies the instance $(p_1 \vee p_2 \vee \neg p_3) \wedge (\neg p_1 \vee p_2 \vee p_3)$. The corresponding $\Gamma \cup \{\alpha > \beta\}$-satisfying hierarchical model in $\mathcal{H}(2)$ is $(\{q_1^+, a_1^1\}, \{q_2^+, a_2^1\}, \{q_3^-, a_3^1\}, \{c^*\})$.

Theorem 7.3: NP**-completeness of PCP for Hierarchical Models**

The $\mathcal{H}(t)$ Preference Consistency Problem is NP-complete for any $t > 1$, even if we restrict the preferences to include only one strict preference statement in $\mathcal{L}_{\geq}^{\mathcal{A}}$.

Proof. Checking whether a hierarchical model satisfies a set of preference statements is polynomial in the number of preference statements and variables, and

thus Preference Consistency is in NP.

Let $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}_{\geq}$ and $\varphi \in \mathcal{L}^{\mathcal{A}}_{\geq}$ be an instance of the Deduction Problem for $\models^{\oplus}_{\mathcal{H}(t)}$ over the hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$. By Theorem 7.1, $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \neg\varphi$ if and only if $\Gamma \cup \{\varphi\}$ is $\mathcal{H}(t)$-consistent. This result gives us an easy reduction from the Deduction Problem to the Consistency Problem. Since, by Theorem 7.2, deciding $\Gamma \not\models^{\oplus}_{\mathcal{H}(t)} \neg\varphi$ is NP-complete, deciding consistency for $\Gamma \cup \{\varphi\}$ is NP-complete. □

7.3 Solving $\mathcal{H}(t)$-Consistency with $t > 1$

In the previous section, we established that $\mathcal{H}(t)$-PDP is coNP-complete and that $\mathcal{H}(t)$-PCP is NP-complete for any $t \geq 2$. A greedy algorithm can solve the special cases $\mathcal{H}(1)$-PCP and $\mathcal{H}(1)$-PDP in time $O(|\mathcal{V}| \cdot |\Gamma|)$ as described in Section 6.3.3. The correctness of this algorithm strongly depends on the fact that all maximal $\Gamma^{(\geq)}$-satisfying $\mathcal{H}(1)$ hierarchical models contain the same variables and (strictly) satisfy the same statements in Γ. This only holds for the class $\mathcal{H}(1)$, and not for the general case of $\mathcal{H}(t)$ as Example 7.5 below shows. In the remainder of this chapter, we concentrate on finding efficient solutions for the NP-complete $\mathcal{H}(t)$-Consistency Problem for $t \geq 2$. In the following, we assume that the operator $\oplus$ is an associative, commutative and strictly monotonic operation ($x \oplus y > z \oplus y$ if $x > z$). Strict monotonicity is explicitly needed in some of the results.

Example 7.5

Suppose as before, a user wants to buy a new prepay mobile phone SIM card and considers different providers based on the price per 10MB data usage (d), the price per text message (m) and the price per minute for calls to the same provider (c). These prices of d, m and c can be combined by the operator$\oplus$ which is the ordinary addition. For any of the four price categories, the lower the price is the better it is for the customer. Consider four different options (providers) α, β, γ and δ with the following prices in cent.

	α	β	γ	δ
d	18	15	13	14
m	15	17	15	13
c	10	11	14	15

Then the model $H = (\{m, c\})$ satisfies $\gamma \prec_H^{\oplus} \beta \equiv_H^{\oplus} \delta \prec_H^{\oplus} \alpha$. That is, under the assumption that the price per text message and the price per minute for calls are the only relevant features and are equally important option α is the best, followed by β and δ which are equally good, and then γ. The model $H'' = (\{c\}, \{m\}, \{d\})$ satisfies $\delta \prec_{H''}^{\oplus} \gamma \prec_{H''}^{\oplus} \beta \prec_{H''}^{\oplus} \alpha$. Thus, under the assumption that the price per minute for calls is more important than the price per text message which is more important than the price per 10MB data usage, option α is the best, followed by β, then γ, and then δ. Both are maximal models of $\Gamma = \{\delta \leq \beta, \gamma \leq \alpha\}$ as they cannot be extended. However, the two models do not include the same variables and H'' satisfies $\delta \leq \beta$ strictly, while H is indifferent between β and δ.

7.3.1 MILP Formulation

We describe a Mixed Integer Linear Programming (MILP) formulation for $\mathcal{H}(t)$-PCP with hierarchical structure $\langle \mathcal{A}, \oplus, \mathcal{V} \rangle$ and preference statements $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$, where the domain D of the variables is integral and $\oplus$ is the standard addition on integers. Let the number of variables in $\mathcal{V}$ be $n = |\mathcal{V}|$. For $\varphi \in \Gamma$, we denote the alternatives involved by α_φ and β_φ, such that φ is either the strict statement $\alpha_\varphi > \beta_\varphi$ or the non-strict statement $\alpha_\varphi \geq \beta_\varphi$. In the case that there exists a feasible solution for the constricted constraints, we can conclude that Γ is consistent.

Assigning Variables to Level Sets: We introduce a matrix of Boolean variables $Y \in \{0, 1\}^{n \times n}$ such that $y_{i,j} = 1$ if and only if variable i is included in the j-th level set of the hierarchical model corresponding to the MIP solution. For $\mathcal{H}(t)$-PCP, every variable is contained in at most one level set and the cardinality of the level sets is bounded by t.

$$\sum_{j=1}^{n} y_{i,j} \leq 1 \quad \text{and} \quad \sum_{j=1}^{n} y_{j,i} \leq t \quad \forall i = 1, \ldots, n. \tag{7.1}$$

Maintaining Values of $\oplus$-combined Level Sets: The matrix of integer vari-

ables $X \in \mathbb{Q}^{n \times |\Gamma|}$ maintains the degree of support/opposition of the statements in the level sets. That is, $x_{j,\varphi} = \bigoplus_{c \in C_j} \alpha_\varphi(c) - \bigoplus_{c \in C_j} \beta_\varphi(c)$ for statement $\varphi \in \Gamma$ and the j-th level set C_j.

$$\sum_{i=1}^{n} y_{i,j}(\alpha_\varphi(c_i) - \beta_\varphi(c_i)) = x_{j,\varphi} \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.2}$$

Next, we define the upper and lower bounds M_φ and m_φ on variables $x_{j,\varphi}$, such that $M_\varphi \geq x_{j,\varphi} \geq m_\varphi$ for all $j = 1, \ldots, n$ and $\varphi \in \Gamma$. These will be used to linearise implication constraints.

$$m_\varphi = \min_{E \subseteq \mathcal{V}} \sum_{c \in E} \alpha_\varphi(c) - \beta_\varphi(c) = \sum_{c \in \mathcal{V}, \alpha_\varphi(c) < \beta_\varphi(c)} \alpha_\varphi(c) - \beta_\varphi(c),$$

$$M_\varphi = \max_{E \subseteq \mathcal{V}} \sum_{c \in E} \alpha_\varphi(c) - \beta_\varphi(c) = \sum_{c \in \mathcal{V}, \alpha_\varphi(c) > \beta_\varphi(c)} \alpha_\varphi(c) - \beta_\varphi(c).$$

Maintaining the Sign of Level Sets (Supporting, Opposing and Indifferent): The Boolean variables $s^{<0}_{j,\varphi}, s^{>0}_{j,\varphi}$ and $s^{=0}_{j,\varphi}$ express the sign for $x_{j,\varphi}$. This is, $s^{<0}_{j,\varphi} = 1$ if and only if $x_{j,\varphi} < 0$, $s^{>0}_{j,\varphi} = 1$ if and only if $x_{j,\varphi} > 0$, and $s^{=0}_{j,\varphi} = 1$ if and only if $x_{j,\varphi} = 0$. Since exactly one of the relations holds,

$$s^{<0}_{j,\varphi} + s^{>0}_{j,\varphi} + s^{=0}_{j,\varphi} = 1 \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.3}$$

To enforce the equivalences between variables $s^{<0}_{j,\varphi}, s^{>0}_{j,\varphi}$, $s^{=0}_{j,\varphi}$ and $x_{j,\varphi}$, we make use of the bounds M_φ and m_φ and the integrity of the variables. In particular, we utilise the fact that the lowest positive value $x_{j,\varphi}$ can take is 1 and the highest negative value is -1. It is enough to enforce three implications. The equivalences then follow by equation 7.3.

For the implication $s^{<0}_{j,\varphi} = 1 \Rightarrow x_{j,\varphi} < 0$, we set the constraint

$$x_{j,\varphi} + s^{<0}_{j,\varphi}(M_\varphi + 1) \leq M_\varphi \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.4}$$

For the implication $s^{>0}_{j,\varphi} = 1 \Rightarrow x_{j,\varphi} > 0$, we set the constraint

$$x_{j,\varphi} + s^{>0}_{j,\varphi}(m_\varphi - 1) \geq m_\varphi \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.5}$$

Finally, we enforce $s_{j,\varphi}^{=0} = 1 \Rightarrow x_{j,\varphi} = 0$ by

$$x_{j,\varphi} - (1 - s_{j,\varphi}^{=0})m_\varphi \geq 0 \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma \text{ and} \tag{7.6}$$

$$x_{j,\varphi} - (1 - s_{j,\varphi}^{=0})M_\varphi \leq 0 \;\; \forall j = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.7}$$

The equivalences, $s_{j,\varphi}^{<0} = 1$ if and only if $x_{j,\varphi} < 0$, $s_{j,\varphi}^{>0} = 1$ if and only if $x_{j,\varphi} > 0$, and $s_{j,\varphi}^{=0} = 1$ if and only if $x_{j,\varphi} = 0$, follow from constraint 7.3 together with constraints 7.4 - 7.7.

Satisfaction of Strict and Non-strict Statements: Following the definition of $\succcurlyeq_H^{\oplus}$, the hierarchical model corresponding to the variable assignments of Y satisfies a non-strict statement φ in Γ if and only if

(I′) for all $i = 1, \ldots, n$, $s_{i,\varphi}^{=0} = 1$; or

(II′) there exists some $i \in \{1, \ldots, n\}$ such that $s_{i,\varphi}^{>0} = 1$ and for all $1 \leq j < i$, $s_{j,\varphi}^{=0} = 1$.

Also, a strict statement φ in Γ is satisfied if and only if (II′) holds.

It is easy to check that conditions (I′) or (II′) hold for all $\varphi \in \Gamma$ if and only if

$$\sum_{j=1}^{i-1} s_{j,\varphi}^{>0} \geq s_{i,\varphi}^{<0} \;\; \forall i = 1, \ldots, n, \forall \varphi \in \Gamma. \tag{7.8}$$

To show this equivalence, assume first that condition (I′) holds for some $\varphi \in \Gamma$. Then $\sum_{j=1}^{i-1} s_{j,\varphi}^{>0} = 0$ and $s_{i,\varphi}^{<0} = 0$ for all $i = 1, \ldots, n$ and $\varphi \in \Gamma$. Now assume that condition (II′) holds for some $\varphi \in \Gamma$, i.e., there exists some $i \in \{1, \ldots, n\}$ such that $s_{i,\varphi}^{>0} = 1$ and for all $1 \leq j < i$, $s_{j,\varphi}^{=0} = 1$. Suppose, there exists $k \in \{1, \ldots, n\}$ such that $s_{k,\varphi}^{<0} = 1$. Then $k > i$. Thus, $\sum_{j=1}^{k-1} s_{j,\varphi}^{>0} \geq 1$ and $s_{k,\varphi}^{<0} = 1$.

Conversely, if Constraint 7.8 holds, then there exists no i and φ for which $s_{i,\varphi}^{>0} = 1$ and $\sum_{j=1}^{k-1} s_{j,\varphi}^{>0} = 0$. Thus, either (I′) or (II′) holds.

Inequality 7.8 yields the satisfaction of $\Gamma^{(\geq)}$. We enforce satisfaction of all strict statements in Γ, by including also the following constraint:

$$\sum_{j=1}^{n} s_{j,\varphi}^{>0} \geq 1 \;\; \forall \varphi \in \Gamma \cap \mathcal{L}_{>}^{\mathcal{A}}. \tag{7.9}$$

Alternative Constraints: The constraints 7.1- 7.9 form a rather simple MILP

formulation for $\mathcal{H}(t)$-PCP. Constraints 7.3- 7.9 could be replaced by sums with extreme weights to enforce a lexicographic order on the level sets. Let $L > 0$ be sufficiently large; then the following two inequalities can be used to replace 7.3-7.9.

$$\sum_{j=1}^{n} \frac{x_{j,\varphi}}{L^j} \geq 0 \quad \forall \varphi \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{\geq}. \tag{7.10}$$

$$\sum_{j=1}^{n} \frac{x_{j,\varphi}}{L^j} > 0 \quad \forall \varphi \in \Gamma \cap \mathcal{L}^{\mathcal{A}}_{>}. \tag{7.11}$$

However, these inequalities can lead to numerical difficulties for a MILP solver. This is true even for small instances with integral variables of small domains and a sophisticated choice for L.

Also, decision variables $y_{i,j}$ could be substituted by $y'_{i,j}$ such that $y'_{i,j} = 1$ if and only if i is included in a level set with index $\geq j$. For each j, the three variables $s^{<0}_{j,\varphi}, s^{>0}_{j,\varphi}, s^{=0}_{j,\varphi} \in \{0, 1\}$ might be replaceable by only one variable $s_{j,\varphi} \in \{0, 1, 2\}$, e.g., so that $s_{j,\varphi} = 0$ corresponds to $s^{<0}_{j,\varphi} = 1$, $s_{j,\varphi} = 1$ corresponds to $s^{>0}_{j,\varphi} = 1$ and $s_{j,\varphi} = 2$ corresponds to $s^{=0}_{j,\varphi} = 1$. However, since our MILP is a satisfaction problem, not an optimization problem, it is not clear whether any of these measures improve the formulation. After trying various Constraint Programming models with set or binary variables, different versions of constraints and different search heuristics, the MILP formulation using inequalities 7.1- 7.9 seemed most promising among this class of approaches.

7.3.2 Recursive Algorithms

In the following, we describe two recursive search algorithms for $\mathcal{H}(t)$-PCP. The algorithms are based on properties of consistency that can be used to prune the search space when searching for a satisfying preference model. Both try to construct a Γ-satisfying hierarchical model by sequentially adding new level sets that do not oppose any preference statement that is not strictly satisfied so far. This implies that, during the algorithm the current model always satisfies $\Gamma^{(\geq)}$, the non-strict version of Γ. We backtrack when the current model cannot be extended further and the model does not satisfy all strict preference statements. The approaches aim to reduce the number of $\Gamma^{(\geq)}$-satisfying hierarchical models constructed by the algorithm. In particular, they try to identify and ignore level sets which cannot lead to a Γ-satisfying hierarchical model although not opposing the preference statements.

Utilising Sequences of Singleton Level Sets: The first approach is based on the idea of including as many singleton level sets as possible in the constructed model. This seems computationally less challenging since a $\Gamma^{(\geq)}$- satisfying sequence of singleton level sets that is maximal in the number of level sets can be found in time $O(|\mathcal{V}| \cdot |\Gamma|)$ (see Section 6.3.3). In the following, we show that for strictly monotonic operators $\oplus$ the recursive search algorithm never needs to backtrack over the choice of such singleton sequences. We first establish the following property for strictly monotonic operators $\oplus$ which can be shown by a short technical proof.

Lemma 7.6. *Let $\oplus$ be an associative, commutative and strictly monotonic operator on the common domain D of the variables $\mathcal{V}$, and let $X, Y \subseteq \mathcal{V}$ be sets of variables with $X \subseteq Y$. Let $\alpha, \beta \in \mathcal{A}$ be alternatives such that X is indifferent under α and β, i.e., $\alpha \equiv_X^{\oplus} \beta$. Then $\alpha \succ_Y^{\oplus} \beta$ if and only if $\alpha \succ_{Y \setminus X}^{\oplus} \beta$. Hence, $\alpha \equiv_Y^{\oplus} \beta$ if and only if $\alpha \equiv_{Y \setminus X}^{\oplus} \beta$.*

Proof. Let $\alpha \succ_Y^{\oplus} \beta$, i.e., $\bigoplus_{c \in Y} \alpha(c) > \bigoplus_{c \in Y} \beta(c)$. Since $\oplus$ is associative and commutative, and $X \subseteq Y$, this is equivalent to $\bigoplus_{c \in Y \setminus X} \alpha(c) \oplus \bigoplus_{c \in X} \alpha(c) > \bigoplus_{c \in Y \setminus X} \beta(c) \oplus \bigoplus_{c \in X} \beta(c)$. By strict monotonicity of $\oplus$ and because X is indifferent under α and β (and thus, $\bigoplus_{c \in X} \alpha(c) = \bigoplus_{c \in X} \beta(c)$), this is equivalent to $\bigoplus_{c \in Y \setminus X} \alpha(c) > \bigoplus_{c \in Y \setminus X} \beta(c)$, i.e., $\alpha \succ_{Y \setminus X}^{\oplus} \beta$. The same argument implies $\alpha \prec_Y^{\oplus} \beta$ if and only if $\alpha \prec_{Y \setminus X}^{\oplus} \beta$. Both equivalences together yield $\alpha \equiv_Y^{\oplus} \beta$ if and only if $\alpha \equiv_{Y \setminus X}^{\oplus} \beta$. □

Note that the previous proof explicitly uses the strict monotonicity of the operator $\oplus$.

Consider the (non-commutative) combination $H \circ' H'$ of two hierarchical models $H = (C_1, \ldots, C_l)$ and $H' = (C'_1, \ldots, C'_k)$ in $\mathcal{H}(t)$ by $(C_1, \ldots, C_l, (C'_1 \setminus \sigma_H), \ldots, (C'_k \setminus \sigma_H))$, where $\sigma_H = \bigcup_{i=1,\ldots,l} C_i$ is used as an abbreviation of the set $\sigma(H)$ of variables involved in the model H. Note that for hierarchical models, level sets are defined to be non-empty, however, sets $C'_i \setminus \sigma_H$ might be empty. We thus define $H \circ H'$ to be the sequence $H \circ' H' = (C_1, \ldots, C_l, (C'_1 \setminus \sigma_H), \ldots, (C'_k \setminus \sigma_H))$ without any empty sets. It is easy to see that $H \circ H'$ is a hierarchical model in $\mathcal{H}(t)$. Furthermore, $\circ$ is a composition operator (see Definition 4.1).

Lemma 7.7. *The operator $\circ : \mathcal{H}(t) \times \mathcal{H}(t) \longrightarrow \mathcal{H}(t)$ on hierarchical models is a composition operator, i.e.,*

1) $H \circ (H' \circ H'') = (H \circ H') \circ H''$, *(associativity)*

2) $H \circ H = H$, *(idempotence)*

3) If $H_1 = H_2 \circ H$ *and* $H_2 = H_1 \circ H'$*, then* $H_1 = H_2$. *(asymmetry)*

for all models $H_1, H_2, H, H', H'' \in \mathcal{H}(t)$.

Proof. Let $H_1, H_2, H, H', H'' \in \mathcal{H}(t)$ be hierarchical models with $H = (C_1, \dots, C_k)$, $H' = (C'_1, \dots, C'_l)$ and $H'' = (C''_1, \dots, C''_m)$. Then $H \circ H' = (C_1, \dots, C_l, (C'_1 \setminus \sigma_H), \dots, (C'_k \setminus \sigma_H))$ without empty sets, and since $1 \leq |C_i| \leq t$ and $1 \leq |C'_i \setminus \sigma_H| \leq |C'_i| \leq t$ (for $C'_i \setminus \sigma_H$ in $H \circ H'$), we have $H \circ H' \in \mathcal{H}(t)$.

Associativity: Consider $H \circ' (H' \circ' H'')$. Then $(H' \circ' H'') = (C'_1, \dots, C'_l, (C''_1 \setminus \sigma_{H'}), \dots, (C''_m \setminus \sigma_{H'}))$. Thus, $H \circ' (H' \circ' H'') = (C_1, \dots, C_k, (C'_1 \setminus \sigma_H), \dots, (C'_l \setminus \sigma_H), (C''_1 \setminus \sigma_{H'}) \setminus \sigma_H, \dots, (C''_m \setminus \sigma_{H'}) \setminus \sigma_H)$. Furthermore, $H \circ' H' = (C_1, \dots, C_k, (C'_1 \setminus \sigma_H), \dots, (C'_l \setminus \sigma_H))$, and thus $(H \circ' H') \circ' H'' = (C_1, \dots, C_k, (C'_1 \setminus \sigma_H), \dots, (C'_l \setminus \sigma_H), (C''_1 \setminus \sigma_{H \circ' H'}), \dots, (C''_m \setminus \sigma_{H \circ' H'}))$. Since $\sigma_{H \circ' H'} = \sigma_H \cup \sigma_{H'}$, $H \circ' (H' \circ' H'') = (H \circ' H') \circ' H''$. Thus, for the composition $\circ$, $H \circ (H' \circ H'') = (H \circ H') \circ H''$.

Idempotence: We have $H \circ H = H$ by definition of $\circ$, since for $H = (C_1, \dots, C_k)$ all sets $C_i \setminus \sigma_H$ are empty and thus left out of the sequence $H \circ H$.

Asymmetry: Suppose $H = H' \circ H_1$ and $H' = H \circ H_2$. Then by definition of $\circ$, $H = (C'_1, \dots, C'_l, \dots)$, i.e., H starts with the sequence of level sets in H'. Similarly, $H' = (C_1, \dots, C_k, \dots)$. Thus, $C_1 = C'_1$, $C_2 = C'_2$, and so on. Hence, $k = l$ and $H = H'$. □

The following proposition shows how the satisfaction of preference statements Γ from H' persists under combination with sequences of singleton level sets that only satisfy $\Gamma^{(\geq)}$.

Proposition 7.8. *Let* $\Gamma \subseteq \mathcal{L}^{\mathcal{A}}$*, and* $\oplus$ *an associative, commutative and strictly monotonic operator on the variable's domain. If* $H = (c_1, \dots, c_l)$ *is a* $\Gamma^{(\geq)}$*- satisfying model in* $\mathcal{H}(1)$ *and* $H' = (C'_1, \dots, C'_k)$ *is a* Γ*-satisfying model in* $\mathcal{H}(t)$*, then* $H \circ H'$ *is a* Γ*-satisfying model in* $\mathcal{H}(t)$.

Proof. We show that $H \circ H'$ satisfies $\Gamma^{(\geq)}$ and strictly satisfies the preference statements that H' strictly satisfies. Hence, $H \circ H'$ is a Γ-satisfying hierarchical model in $\mathcal{H}(t)$.

Recall that a preference statement φ is strictly satisfied when there exists a level set C supporting φ, i.e., $\alpha_\varphi \succ_C \beta_\varphi$, and all preceding level sets C' are indifferent under φ, i.e., $\alpha_\varphi \equiv_{C'} \beta_\varphi$. Hence, the preference statements in Γ that

are strictly satisfied by $H = (c_1, \ldots, c_l)$ are also strictly satisfied by $H \circ H' = (c_1, \ldots, c_l, (C'_1 \setminus \sigma_H), \ldots, (C'_k \setminus \sigma_H))$. Let Γ' be the set of remaining preference statements that are not strictly satisfied by H.

Since H satisfies $\Gamma^{(\geq)}$, H is indifferent under all statements in $\varphi \in \Gamma'$, i.e., $\alpha_\varphi(c_i) = \beta_\varphi(c_i)$ for all $1 \leq i \leq l$. Consider an arbitrary level set C in H' and a preference statement $\varphi \in \Gamma'$. Repeatedly applying Lemma 7.6 for the singleton level sets in $\sigma_H \cap C$ in connection with a level set C in H' yields: $\alpha_\varphi \sim^{\oplus}_{C} \beta_\varphi$ if and only if $\alpha_\varphi \sim^{\oplus}_{C \setminus \sigma_H} \beta_\varphi$, where $\sim$ is one of the relations $\succ$, $\equiv$ or $\prec$. Thus, the level sets $C'_i \setminus \sigma_H$ in $H \circ H'$ have the same relation towards statements $\varphi \in \Gamma'$ as the level sets C'_i in H'. Since the initial singleton sequence in $H \circ H'$ is indifferent under preference statements $\varphi \in \Gamma'$, $H \circ H'$ satisfies φ if and only if H' satisfies φ. Also, all statements $\Gamma \setminus \Gamma'$ are strictly satisfied by $H \circ H'$. Hence, $H \circ H'$ satisfies $\Gamma^{(\geq)}$ and strictly satisfies all statements in Γ that H' strictly satisfies. Therefore, $H \circ H'$ satisfies Γ. □

The statement of Proposition 7.8, can be seen as a weak version of strong compositionality of preference statements $\mathcal{L}^{\mathcal{A}}$ for the composition operator $\circ$ on models $\mathcal{H}(t)$. The Γ-satisfaction of a model is preserved under combination with a $\models^*$ model of Γ in $\mathcal{H}(1)$. In general statements in $\mathcal{L}^{\mathcal{A}}$ are not strongly compositional for models $\mathcal{H}(t)$, and in fact non-strict statements are not even compositional as the following example shows.

Example 7.6

Consider variables $\mathcal{V} = \{c, d, e, f\}$ expressing ratings for cost, distance, experience and food quality of restaurants $\mathcal{A} = \{\alpha, \beta\}$. Let the operator $\oplus$ be the ordinary addition on the integers by which the ratings can be combined. Assume the higher the rating is, the better the alternative with respect to the corresponding feature.

	α	β
c	3	5
d	5	3
e	1	2
f	2	1

In this context, the two model $H = (\{c, d\})$ and $H' = (\{d, e\})$ both satisfy the non-strict statement $\alpha \geq \beta$ in $\mathcal{L}^{\mathcal{A}}$. That is, no matter whether cost and distance are considered together or experience and distance, the restaurant α is preferred to β. However, the composition $H \circ H' = (\{c, d\}, \{e\})$, which incorporates the second viewpoint into the first, satisfies $\alpha < \beta$, and thus does not satisfy $\alpha \geq \beta$. Hence, $\alpha \geq \beta$ is not generally compositional under addition for hierarchical models $\mathcal{H}(t)$ with $t > 1$. Furthermore, $H \models^* \alpha > \beta$, since the extension $(\{c, d\}, \{f\})$ satisfies the strict statement. Also, $H' \models \alpha > \beta$. However, the composition $H \circ H' = (\{c, d\}, \{e\})$ satisfies $\alpha < \beta$. Hence, $\alpha > \beta$ is not strongly compositional under addition for hierarchical models $\mathcal{H}(t)$ with $t > 1$.

Proposition 7.8 immediately leads to the next result.

Theorem 7.4: Singleton Set Sequences in Γ-Satisfying Models

Let H be a maximal $\mathcal{H}(1)$-model of $\Gamma^{(\geq)}$, i.e., $H \in \mathcal{H}(1)$ satisfies $\Gamma^{(\geq)}$ and cannot be extended by another singleton level set without opposing some statement in Γ. If Γ is $\mathcal{H}(t)$-consistent, then there exists a Γ-satisfying model in $\mathcal{H}(t)$ with H as initial sequence.

Proof. Suppose Γ is $\mathcal{H}(t)$-consistent. Then there exists a model H' of Γ. By Proposition 7.8, $H \circ H'$ satisfies Γ. Also, $H \circ H'$ has H as initial sequence. □

Based on Theorem 7.4, we describe the algorithm PC-check($\mathcal{V}, \Gamma, \oplus, t,$) that solves $\mathcal{H}(t)$-PCP by trying to construct a Γ-satisfying hierarchical model. This method is summarised in the algorithm below. After finding an initial singleton sequence $(c_1, \ldots, c_k)$ that is maximal while satisfying $\Gamma^{(\geq)}$ (in time $O(|\mathcal{V}| \cdot |\Gamma|)$ by a greedy algorithm, see Section 6.3.3), we consider possible (non-opposing) level sets C of size $2 \leq |C| \leq t$. Let Γ' be the set of preference statements in Γ that are not strictly satisfied by $H = (c_1, \ldots, c_k, C)$. We try to extend the sequence H by another Γ'-satisfying hierarchical model. We construct this extending model by recursively calling the method for the subproblem with statements Γ' and variables $\mathcal{V}' = \mathcal{V} - \{c_1, \ldots, c_k\} - C$. If no such extension exists (that satisfies Γ), we backtrack over the last chosen level set C and try a new level set. Note that by Theorem 7.4, we never have to backtrack over the choice of singleton level sets, which can be a significant advantage over solving the MILP model. As soon as the currently considered sequence in the algo-

rithm satisfies Γ, we stop and return the sequence, showing that the instance is $\mathcal{H}(t)$-consistent. If no Γ-satisfying sequence can be found after exploiting all possible ($\Gamma^{(\geq)}$-satisfying) hierarchical models, we stop and return an empty sets signifying that the instance is $\mathcal{H}(t)$-inconsistent.

Algorithm 7.1: PC-check($\mathcal{V}, \Gamma, \oplus, t,$)

```
1.  H ← (c_1, ..., c_k) some maximal H(1)-model of Γ^(≥)
2.  IF ( H ⊨ Γ ) THEN
3.       RETURN H and STOP.
4.  FOR ALL C ⊆ V − {c_1, ..., c_k} with 2 ≤ |C| ≤ t and
            α_φ ≻_C^⊕ β_φ for all φ ∈ Γ with α_φ ≡_H^⊕ β_φ DO
5.      H' ← (c_1, ..., c_k, C)
6.      IF ( H' ⊨ Γ ) THEN
7.          RETURN H' and STOP.
8.      ELSE Γ' = {φ ∈ Γ | α_φ ≡_H'^⊕ β_φ}
9.           V' = V − σ(H')
10.          H'' ← H' ∘ PC-check(V', Γ', ⊕, t,)
11.          IF ( H'' ⊨ Γ ) THEN
12.              RETURN H'' and STOP.
13. RETURN ∅ and STOP.
```

Maintaining Conflicting Sets: In the following, we extend the algorithm PC-check($\mathcal{V}, \Gamma, \oplus, t$) by maintaining conflicting sets that cannot be contained in the later level sets, and thus reduce the number of backtracks. Proposition 7.9 shows that the satisfaction of Γ persists in a certain case for a hierarchical model H' when combining with a hierarchical model H that extends an initial sequence of level sets of H' by one more level set and only satisfies $\Gamma^{(\geq)}$.

Proposition 7.9. *Let $(\langle \mathcal{A}, \oplus, \mathcal{V}\rangle, \Gamma)$ be an instance of $\mathcal{H}(t)$-PCP that is $\mathcal{H}(t)$-consistent. Let $H = (C_1, \ldots, C_k, B)$ be a $\Gamma^{(\geq)}$-satisfying model in $\mathcal{H}(t)$, and let $H' = (C_1, \ldots, C_k, C_{k+1}, \ldots, C_l)$ be a Γ-satisfying hierarchical model in $\mathcal{H}(t)$ with $B \subseteq C_j$ for some $k+1 \leq j \leq l$. Then $H \circ H' = (C_1, \ldots, C_k, B, C_{k+1}, \ldots, (C_j \setminus B), \ldots, C_l)$ is a Γ-satisfying hierarchical model in $\mathcal{H}(t)$.*

Proof. We show that the model $H \circ H' = (C_1, \ldots, C_k, B, C_{k+1}, \ldots, C_{j-1}, C_j \setminus B, C_{j+1}, \ldots, C_l)$ satisfies $\Gamma^{(\geq)}$ and strictly satisfies all statements that are strictly

satisfied by H'. Hence, $H \circ H'$ is a Γ-satisfying hierarchical model in $\mathcal{H}(t)$.

A preference statement φ is strictly satisfied when there exists a level set C supporting φ, i.e., $\alpha_\varphi \succ_C \beta_\varphi$, and all preceding level sets C' are indifferent under φ, i.e., $\alpha_\varphi \equiv_{C'} \beta_\varphi$. Hence, the preference statements in Γ that are strictly satisfied by $H = (C_1, \dots, C_k, B)$ are also strictly satisfied by $H \circ H'$. In the following, we consider all remaining preference statements. Let Γ' be the set of preference statements in Γ that are not strictly satisfied by H.

Since H satisfies $\Gamma^{(\geq)}$, the sequence $(C_1, \dots, C_k, B)$ is indifferent under all statements in Γ', i.e., $\bigoplus_{c \in C} \alpha_\varphi(c) = \bigoplus_{c \in C} \beta_\varphi(c)$ for all $\varphi \in \Gamma'$ and $C \in \{C_1, \dots, C_k, B\}$. The level sets C_i with $k+1 \leq i \leq l$ and $i \neq j$ are level sets in both H' and $H \circ H'$. Hence, for the satisfaction of statements in Γ', we only need to compare the level set C_j in H' to the level set $C_j \setminus B$ in $H \circ H'$. Consider a preference statement $\varphi \in \Gamma'$. Since B is indifferent under φ, by Lemma 7.6, $\alpha_\varphi \sim^{\oplus}_{C_j} \beta_\varphi$ if and only if $\alpha_\varphi \sim^{\oplus}_{C_j \setminus B} \beta_\varphi$, where $\sim$ is one of the relations $\succ$, $\equiv$ or $\prec$. Thus, for $\varphi \in \Gamma'$ all level sets $C_{k+1}, \dots, C_l$ in H' have the same relation towards φ as the level sets $C_{k+1}, \dots, C_{j-1}, C_j \setminus B, C_{j+1}, \dots, C_l$ in $H \circ H'$. Since the initial sequence $(C_1, \dots, C_k, B)$ is indifferent under preference statements $\varphi \in \Gamma'$, $H \circ H'$ satisfies φ if and only if H' satisfies φ. Furthermore, all statements $\Gamma \setminus \Gamma'$ are strictly satisfied by $H \circ H'$. We have shown that $H \circ H'$ satisfies $\Gamma^{(\geq)}$ and strictly satisfies all preference statements that H' strictly satisfies. □

Reformulating Proposition 7.9 yields the following statement.

Theorem 7.5: Conflicting Sets

Let $H = (C_1, \dots, C_k, B)$ be a $\Gamma^{(\geq)}$-satisfying hierarchical model in $\mathcal{H}(t)$. If there exists no extension $(C_1, \dots, C_k, B, C_{k+1}, \dots, C_l)$ of H in $\mathcal{H}(t)$ that satisfies Γ, then for all Γ-satisfying hierarchical models $H' = (C_1, \dots, C_k, C_{k+1}, \dots, C_l)$ in $\mathcal{H}(t)$, we have $B \nsubseteq C_j$ for all $k+1 \leq j \leq l$.

This proposition characterises the conflicting sets B that are maintained in the second recursive approach. By Theorem 7.5, no Γ-satisfying hierarchical model that extends $(C_1, \dots, C_k)$ can contain the conflicting set B. Thus, at a point of the algorithm where we backtrack because no Γ-satisfying extension of the current hierarchical model can be found, we add the last considered level set to the list of conflicting sets. We then choose a new next level set that does not contain any conflicting set. This extension of the algorithm PC-check$(\mathcal{V}, \Gamma, \oplus, t)$

is given as the algorithm PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S} = \emptyset, s$) below. Here, although reducing the search space, we have to maintain a list of conflicting sets which can grow exponentially large. Thus, it is not obvious if maintaining conflicting sets is advantageous. We introduce the additional parameter s which is a cardinality bound on the size of the conflicting sets. Only conflicting sets C with $|C| \leq s$ are maintained, so that the space needed is $O(s \cdot \binom{n}{s})$.

Algorithm 7.2: PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S} = \emptyset, s$)

```
1.  H ← (c_1, ..., c_k) some maximal H(1)-model of Γ^(≥)
2.  IF ( H ⊨ Γ ) THEN
3.       RETURN H and STOP.
4.  FOR ALL C ⊆ V − {c_1, ..., c_k} with 2 ≤ |C| ≤ t and
            α_φ ≽^⊕_C β_φ for all φ with α_φ ≡^⊕_H β_φ
            such that there ∄ S ∈ S with S ⊆ C DO
5.      H' ← (c_1, ..., c_k, C)
6.      IF ( H' ⊨ Γ ) THEN
7.           RETURN H' and STOP.
8.      ELSE Γ' = {φ ∈ Γ | α_φ ≡^⊕_H' β_φ}
9.           V' = V − σ(H')
10.          H'' ← H' ∘ PC-check(V', Γ', ⊕, t, S, s)
11.          IF ( H'' ⊨ Γ ) THEN
12.               RETURN H'' and STOP.
13.          ELSE IF ( |C| ≤ s ) THEN
14.                    S ← S ∪ {C}
15. RETURN ∅ and STOP.
```

Example 7.7

Consider a $\mathcal{H}(3)$-PCP instance with the following variables $c_1, \ldots, c_5$ and statements on the alternatives $\alpha, \beta, \gamma, \delta$ given by $\Gamma = \{\alpha \leq \beta, \beta \leq \gamma, \gamma < \delta\}$.

	α	$\leq$	β	$\leq$	γ	$<$	δ
c_1	1		0		0		0
c_2	0		2		2		2
c_3	1		1		0		1
c_4	0		2		1		1
c_5	2		0		1		0

Let $\oplus$ be the standard addition on integers. Suppose, in the first step PC-check finds the maximal singleton sequence (c_2, c_1) (which cannot be extended by any other variable without violating $\Gamma^{(\geq)}$). Then the algorithm will in turn consider sets $\{c_3, c_4\}$, $\{c_3, c_5\}$, $\{c_4, c_5\}$ and $\{c_3, c_4, c_5\}$. The sequences $c_2, c_1, \{c_3, c_4\}$ and $c_2, c_1, \{c_4, c_5\}$ violate $\Gamma^{(\geq)}$. The sequence $c_2, c_1, \{c_3, c_5\}$ satisfies $\Gamma^{(\geq)}$ but cannot be extended to satisfy Γ. In PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$), the set $\{c_3, c_5\}$ is added to the conflicting sets $\mathcal{S}$ and thus the set $\{c_3, c_4, c_5\}$ does not have to be checked (by Proposition 7.9). PC-check($\mathcal{V}, \Gamma, \oplus, t$) finds that $c_2, c_1, \{c_3, c_4, c_5\}$ violates $\Gamma^{(\geq)}$. Thus none of the possible extending sets leads to a Γ-satisfying sequence and "Inconsistent" is returned. Note that PC-check does not have to backtrack over the choice of variables in the initial singleton set sequence c_2, c_1 (by Theorem 7.4).

7.4 Experimental Runtime Comparisons

In our experiments, we compare the approaches from Section 7.3.1 and 7.3.2 for solving PCP by their running time. Here, the MILP formulation functions as a baseline and is expected to be outperformed by the two recursive approaches as they directly exploit the problem structure to perform less backtracks in a way that is not recognized by the CPLEX solver that we use to solve the MILP formulation. Note that it is not obvious how to incorporate the pruning of the search space that is performed by the recursive algorithms, in a MILP model in the form of constraints or heuristics (if indeed it is possible at all). We investigate the degree of improvement of the recursive algorithms towards the rather sim-

ple MILP formulation and the relation of the recursive algorithms towards each other. Though PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) prunes the search space further than PC-check($\mathcal{V}, \Gamma, \oplus, t$), the list of maintained conflicting sets can grow extremely large. Thus, it is not obvious if maintaining conflicting sets is advantageous.

Instances: For our experiments, we considered different instance sizes in order to observe the effect on the running time by varying the number of variables n and the number of preference statements g. For the lack of real world data, we generated 50 instances uniformly at random with variables each with domain $\{0, 1, 2, 3, 4, 5\}$ for each of the problem sizes $n \in \{10, 15, \ldots, 35\}$ and $g \in \{10, 15, \ldots, 50\}$, where we fix the number of alternatives that the preference statements are based on to $m = 25$.

First an $n \times m$ matrix is generated that gives the values of the variables for the alternatives. We next draw g ordered pairs of alternatives (α_i, α_j) with $i < j$ uniformly at random (without repetition) such that the corresponding preference statement $\alpha_i \geq \alpha_j$ or $\alpha_i > \alpha_j$ coincides with the linear order $\alpha_1 > \cdots > \alpha_m$. This way, we avoid cycles in the statements, which trivially lead to inconsistency. The first $\lceil g/2 \rceil$ statements are handles as strict statements, the remaining statements are non-strict statements. Note that not all alternatives generated are involved in preference statements. Thus, m does not have a direct influence on the size of the search space or the running time.

Implementation: We implemented all three approaches in Java Version 1.8 using the IBM ILOG CPLEX (version 12.6.2) library for the MILP formulation. All experiments were conducted independently on a 2.66Ghz quad-core processor with 12GB memory.

We choose $\oplus$ as the standard addition on the integers as in Section 7.3.1. To reduce the number of experiments, we allow the cardinality bound on the level sets to be $t = n$, the number of variables, and fix the cardinality bound on the maintained conflicting sets to $s = 5$ (which gives the bound $|\mathcal{S}| \leq \binom{n}{s} \leq \binom{35}{5} = 324632$). Since $\mathcal{H}(k') \subseteq \mathcal{H}(k)$ for all $k' < k$, we expect that the running times would be lower for smaller t. Also, $\mathcal{H}(k) = \mathcal{H}(n)$ for all $k \geq n$, i.e., the running times are the same for bigger t. For the recursive algorithms, we enumerate the next level sets with lower cardinality before ones with higher cardinality, and level sets containing variables with smaller indexes before ones with higher indexes.

Experimental Results: As expected, solving the MILP formulation of PCP (as

presented in Section 7.3.1) by the CPLEX solver is much slower than by the two recursive algorithms PC-check (as presented in Section 7.3.2), see Table 7.1. However, it is remarkable how quickly the ratio between the mean times of solving the MILP and PC-check grows with the number of statements and variables in the instances.

Table 7.1: Mean running times in seconds to solve PCP with the MILP formulation over 25 instances and with PC-check over 50 instances.

$g =$		$n = 10$	$n = 15$	$n = 20$
10	PC-check$(\mathcal{V}, \Gamma, \oplus, t)$	0.011	0.01	0.04
	PC-check$(\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s)$	0.003	0.01	0.03
	MILP	0.22	10.53	149.81
	ratio: MILP/PC-check	≥ 20	≥ 1053	≥ 3745.25
15	PC-check$(\mathcal{V}, \Gamma, \oplus, t)$	0.003	0.01	0.29
	PC-check$(\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s)$	0.001	0.01	0.28
	MILP	0.22	220.28	>41472
	ratio: MILP/PC-check	≥ 73.33	≥ 22028	≥ 143006.89

The two algorithms PC-check$(\mathcal{V}, \Gamma, \oplus, t)$ and PC-check$(\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s)$ show similar behaviour of running times for different instance sizes. Figure 7.1 shows some of the running times for PC-check$(\mathcal{V}, \Gamma, \oplus, t)$ and PC-check$(\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s)$. Here, we can see that the running times increase with the number of variables n and the number of statements g.

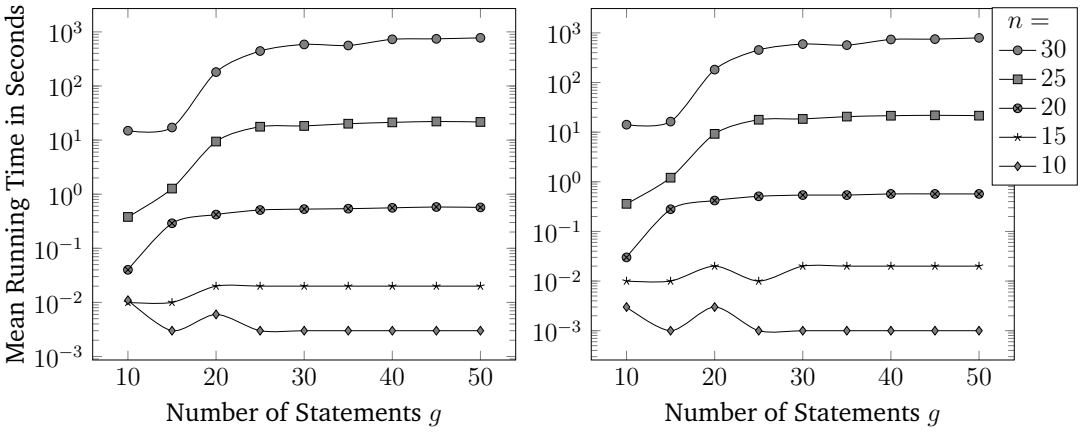


Figure 7.1: Mean running times in seconds of PC-check$(\mathcal{V}, \Gamma, \oplus, t)$ (left) and PC-check$(\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s)$ (right).

For time reasons experiments were only run for 5 instances (instead of 50) for some larger instance sizes. This may explain the irregularity at $n = 35, g = 30$. Table 7.2 shows all running times for algorithms PC-check.

Table 7.2: Mean times of PC-check in seconds fixing $m = 25$ and ratios of the mean times between PC-check($\mathcal{V}, \Gamma, \oplus, t$) and PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) rounded to the nearest hundredth. *Mean time over 5 instances only. (All remaining are mean times over 50 instances.)

$g =$		$n = 10$	$n = 15$	$n = 20$	$n = 25$	$n = 30$	$n = 35$
10	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.011	0.01	0.04	0.38	14.9	0.005
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.003	0.01	0.03	0.36	14.16	0.003
	ratio	3.67	1	1.33	1.06	1.05	1.67
15	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.01	0.29	1.27	17.1	522.26
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.01	0.28	1.21	16.34	500.13
	ratio	3	1	1.04	1.05	1.05	1.04
20	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.006	0.02	0.42	9.41	180.83	2165.83
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.003	0.02	0.42	9.25	182.09	2159.93
	ratio	2	1	1	1.02	0.99	1.00
25	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.51	17.67	442.79	5393.47*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.01	0.51	17.81	452.74	5377.27*
	ratio	3	2	1	0.99	0.98	1.00
30	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.53	18.27	586.16	6.57*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.02	0.54	18.47	595.09	6.32*
	ratio	3	1	0.98	0.99	0.98	1.04
35	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.54	20.09	560.51	16796.17*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.02	0.54	20.6	567.16	16503.98*
	ratio	3	1	1	0.98	0.99	1.02
40	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.56	21.24	729	24494.72*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.02	0.57	21.45	736.87	24269.56*
	ratio	3	1	0.98	0.99	0.99	1.01
45	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.58	22.08	744.17	23180.91*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.02	0.57	21.86	749.54	22886.84*
	ratio	3	1	1.02	1.01	0.99	1.01
50	PC-check($\mathcal{V}, \Gamma, \oplus, t$)	0.003	0.02	0.57	21.62	776	27084.24*
	PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$)	0.001	0.02	0.57	21.51	795	26955.88*
	ratio	3	1	1	1.01	0.98	1.00

A detailed analysis shows that in 57% of the measured instance sizes PC-check($\mathcal{V}, \Gamma, \oplus, t$) is slower than PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$). The ratios of the mean running times of the two algorithms demonstrate that PC-check($\mathcal{V}, \Gamma, \oplus, t$) is at most 3.67 times slower than PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) and PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S}, s$) is at most 1.03 times slower than PC-check($\mathcal{V}, \Gamma, \oplus, t$). This small difference between the running times of the two algorithms indicates that for these instances reducing the number of backtracks by ruling out conflicting sets (of size < 5) is not very worthwhile.

Table 7.2 indicates that the running times for PC-check increase with increasing number of preference statements. Also, the running times tend to increase for increasing number of variables (with some exceptions).

Interpretation of Experimental Results Exploiting the theoretical results on properties of consistent instances developed in Section 7.3.2 allow the algorithms PC-check to prune the search space much further than a MILP solver could do for the MILP formulation given in Section 7.3.1. The experimental results confirm that the algorithms PC-check are solving the instances faster than CPLEX. Even more, the ratios between the mean solving times of the MILP and PC-check increase extremely quickly with the number of variables and statements. It is not obvious how the pruning rules of the PC-check algorithms could be incorporated in the MILP formulation as constraints.

There is relatively very little difference between the mean running times of the two recursive algorithms PC-check on the tested instances. Thus, in PC-check($\mathcal{V}, \Gamma, \oplus, t, \mathcal{S} = \emptyset, s$), the effort of maintaining a list $\mathcal{S}$ of (possibly exponentially many) conflicting sets to prune the search space further, is not strongly paying off.

7.5 Discussion

In this chapter, we established that the Deduction and Consistency Problem for hierarchical models $\mathcal{H}(t)$ with $t > 1$ with preference statements $\mathcal{L}^{\mathcal{A}}$ are NP-complete and coNP-complete, respectively, even if one restricts the cardinality of the equal-importance sets of variables to have at most two elements.

However, the special case where a fixed equivalence relation on variables is given that specifies the possible level sets is polynomial time solvable by applying the algorithm from Section 6.3.3.

We developed a Mixed Integer Linear Program formulation for the $\mathcal{H}(t)$-Consistency Problem, and then approach the problem with two variants of a recursive search that rely on pruning rules of the search space. The first recursive search tries to find a satisfying model that includes as many singleton sets as possible, since they can be found in polynomial time. The second variant of the recursive search, extends the first approach by additionally maintaining a list of conflicting sets, which can not be included in extending models. Our runtime experiments show, as expected, that the recursive search approaches which explicitly exploit the problem structure, outperform the MILP solver. However, they also indicate that maintaining a large list of conflicting sets does not improve the runtime of the recursive search significantly.

To find an explanation for the behaviour of the running times, we could observe the occurrence of instances that have solutions in $\mathcal{H}(t)$ with $t > 1$, in $\mathcal{H}(1)$, or are $\mathcal{H}(t)$-inconsistent. The whole search space must be explored until deciding inconsistency for $\mathcal{H}(t)$, which can lead to high running times. In contrast, PC-check solves $\mathcal{H}(1)$-consistent instances in polynomial time. The instance distribution might suggest that the running times go up with the number of inconsistent instances. A further analysis of the experiments could involve the size of the search space, i.e., counting the number of $\Gamma^{(\geq)}$-satisfying hierarchical models and the number of hierarchical models that were actually considered during the search. Also, one could try using a relaxation of a MILP formulation as a fast check for inconsistency within PC-check$(\mathcal{V}, \Gamma, \oplus, t)$. If the relaxation shows that the current subproblem is inconsistent, we can avoid another (time consuming) recursive call.

Chapter 8

CVO Lexicographic Model

In this chapter, we analyse the problems of consistency and inference based on cvo lexicographic models $\mathscr{L}$ for comparative preference languages $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$. For better readability, we will drop the annotation "cvo" in most places in this chapter. They include forms of the statements $\phi^{\mathcal{R}}$ from Section 4.3.2.2, where $\mathcal{R}$ is a set of pairs of alternatives. In many natural situations, $\mathcal{R}$ can be exponentially large; in the languages discussed here, we are able to express certain exponentially large sets $\mathcal{R}$ compactly. We will see that even for these general preference languages, cvo lexicographic models allow for efficient algorithms to solve consistency and inference.

The method introduced in Section 8.1 is a detailed description of the general algorithm formulated in the previous chapter in Section 4.2.4. We previously established the strong compositionality of statements in languages $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ with respect to cvo lexicographic models $\mathscr{L}$ (see Theorem 4.2). Other preliminary results for $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$ in connection with $\mathscr{L}$ are discussed in the beginning of Section 8.1. The general algorithm uses a greedy approach which consists of repeatedly finding minimal extensions that do not oppose any preference statement. Conditions for Γ-satisfaction (for $\models^*$-models of Γ) are developed in Section 8.1.2. In Section 8.1.2 we characterise minimal extensions for cvo lexicographic models $\mathscr{L}$ and outline how they can be found. Section 8.1.4 summarises the previous results in a formal description of the algorithm.

Related work on $\mathscr{L}$ models considers preference inference and develops an efficient algorithm similar to ours for the case where preference statements are restricted to be only non-strict statements $p \geq q \mid T$ in $\mathcal{L}_{pqT}$, [Wil14]. This is again a greedy approach that aims at finding a maximal model of Γ, how-

ever, since the preference statements are only non-strict statements in $\mathcal{L}_{pqT}$, the conditions of finding extending models are somewhat simpler.

We describe different notions of optimality in Section 8.2.1 and analyse these for the case of CVO lexicographic models $\mathscr{L}$ and compositional statements in Section 8.2.2. A detailed analysis of computational methods and complexities for the case of models $\mathscr{L}$ and statements $\mathcal{L}'_{pqT}$ is provided in Section 8.2.3. We end the chapter with a brief discussion.

Most parts of this chapter originate from [WG17] and some from [GRW15].

8.1 $\mathscr{L}$-Consistency for $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$

Recall from Definition 3.5 that the language $\mathcal{L}_{pqT}$ consists of all preference statements of the form $p \rhd q \mid T$, where $\rhd$ is either $\geq$, or $\gg$ or $>$, and P, Q and T are subsets of V, with $(P \cup Q) \cap T = \emptyset$, and $p \in \underline{P}$ is an assignment to P, and $q \in \underline{Q}$ is an assignment to Q. Here, the statement $p \rhd q \mid T$ represents that p is preferred to q if T is held constant.

Statements of the form $p \geq q \mid T$ are called *non-strict*; statements of the form $p \gg q \mid T$, are called *fully strict*, and statements of the form $p > q \mid T$ are called *weakly strict*.

For any statement $\phi \in \mathcal{L}_{pqT}$ equalling $p \rhd q \mid T$, the set ϕ^* is defined as the set of tuples of alternatives (α, β), such that α extends the partial assignments p and β extends q, and α and β agree on all variables in T (see Definition 3.6). $\phi^{(\geq)}$ is defined to be $p \geq q \mid T$, the *non-strict version of* ϕ. For lex model π, we define:

- π satisfies $\phi^{(\geq)}$, if $\alpha \succcurlyeq_\pi \beta$ for all $(\alpha, \beta) \in \phi^*$.
- π satisfies fully strict ϕ, if $\alpha \succ_\pi \beta$ for all $(\alpha, \beta) \in \phi^*$.
- π satisfies weakly strict ϕ, if π satisfies $\phi^{(\geq)}$ and if $\alpha \succ_\pi \beta$ for some $(\alpha, \beta) \in \phi^*$.

For alternatives α and β, a non-strict preference of α over β can be represented as $\alpha \geq \beta \mid \emptyset$, which is equivalent to the non-strict preference statement $\alpha \geq \beta$ in $\mathcal{L}^{\mathcal{A}}$, so we abbreviate it to that. Similarly, we abbreviate $\alpha > \beta \mid \emptyset$ to $\alpha > \beta$ (which is also equivalent to $\alpha \gg \beta \mid \emptyset$).

We can write a statement $\phi \in \mathcal{L}_{pqT}$ as $ur \rhd us \mid T$, where $u \in \underline{U}$, $r \in \underline{R}$, $s \in \underline{S}$, and U, T and $R \cup S$ are (possibly empty) mutually disjoint subsets of V, and

for all $X \in R \cap S$, $r(X) \neq s(X)$. For such a representation, we write $u_\phi = u$, $r_\phi = r$, $s_\phi = s$, $U_\phi = U$, $R_\phi = R$, $S_\phi = S$ and $T_\phi = T$. We assume, without loss of generality, that for $X \in \mathcal{V}$, if $|\underline{X}| = 1$ then $X \in T_\phi$. This ensures that such a representation is unique. We also define $W_\phi = V \setminus (R_\phi \cup S_\phi \cup T_\phi \cup U_\phi)$.

8.1.1 Projections to Y

Recall the definition of $(\phi^*)^{\downarrow Y}_A$, the A-restricted projection of statement $\phi \in \mathcal{L}_{pqT}$ to Y, from Definition 4.18. $(\phi^*)^{\downarrow Y}_A$, for $Y \in \mathcal{V}$ and $A \subseteq \mathcal{V} - \{Y\}$, is the set of pairs $(\alpha(Y), \beta(Y))$ such that $(\alpha, \beta) \in \phi^*$ and $\alpha(A) = \beta(A)$. For a comparative preference statement ϕ we abbreviate $(\phi^*)^{\downarrow Y}_A$ to $\phi^{\downarrow Y}_A$.

Proposition 1 of [Wil14] leads to the following result.

Proposition 8.1. *Consider any element $\phi \in \mathcal{L}_{pqT}$ written as the unique representation $u_\phi r_\phi \rhd u_\phi s_\phi \mid T_\phi$, where $u_\phi \in \underline{U_\phi}$, $r_\phi \in \underline{R_\phi}$, $s_\phi \in \underline{S_\phi}$, and for all $X \in R_\phi \cap S_\phi$, $r_\phi(X) \neq s_\phi(X)$. Let A be a set of variables and let Y be a variable not in A.*

If $R_\phi \cap S_\phi \cap A \neq \emptyset$ then $(\phi^*)^{\downarrow Y}_A$ is empty. Otherwise, $(\phi^*)^{\downarrow Y}_A$ consists of all pairs $(y, y') \in \underline{Y} \times \underline{Y}$ such that (i) $y = y'$ if $Y \in T_\phi$; (ii) $y = y' = u_\phi(Y)$ if $Y \in U_\phi$ (iii) $y = r_\phi(Y)$ if $Y \in R_\phi$; and (iv) $y' = s_\phi(Y)$ if $Y \in S_\phi$. Thus if $R_\phi \cap S_\phi \cap A = \emptyset$ and $Y \in W_\phi$ then $(\phi^*)^{\downarrow Y}_A = \underline{Y} \times \underline{Y}$.

Proof. First suppose $R_\phi \cap S_\phi \cap A \neq \emptyset$. Then there exists a variable $X \in R_\phi \cap S_\phi \cap A$ with $r_\phi(X) \neq s_\phi(X)$, by our definition of sets R_ϕ and S_ϕ. Thus, there does not exist a pair of alternatives (α, β) extending $u_\phi r_\phi$ and $u_\phi s_\phi$, respectively, such that $\alpha(X) = \beta(X)$ for the variables $X \in R_\phi \cap S_\phi \cap A$. Hence, $(\phi^*)^{\downarrow Y}_A$ is empty.

Assume $R_\phi \cap S_\phi \cap A = \emptyset$ for the remainder of the proof.

If $Y \in T_\phi$, then all tuples in $(\alpha, \beta) \in (\phi^*)_A$ satisfy $\alpha(T_\phi) = \beta(T_\phi)$, and thus $y = y'$ for all pairs $(y, y') \in \underline{Y} \times \underline{Y}$. For $(\alpha, \beta) \in (\phi^*)_A$ and any $y \in \underline{Y}$, we can define $\alpha'(X) = \alpha(X)$ and $\beta'(X) = \beta(X)$ for all $X \in \mathcal{V} \setminus \{Y\}$, and $\alpha'(Y) = \beta'(Y) = y$. Then $(\alpha', \beta') \in (\phi^*)_A$ and hence, for any $y \in \underline{Y}$, we have $(y, y) \in (\phi^*)^{\downarrow Y}_A$.

Similarly, if $Y \in U_\phi$, then all tuples in $(\alpha, \beta) \in (\phi^*)_A$ satisfy $\alpha(U_\phi) = \beta(U_\phi) = u_\phi$, and thus $y = y' = u_\phi(Y)$.

For the case that $Y \in R_\phi$, all tuples in $(\alpha, \beta) \in (\phi^*)_A$ satisfy $\alpha(R_\phi) = r_\phi$, and thus $y = r_\phi(Y)$. For $(\alpha, \beta) \in (\phi^*)_A$ and any $y \in \underline{Y}$, we can define $\beta'(X) = \beta(X)$

for all $X \in \mathcal{V} \setminus \{Y\}$, and $\beta'(Y) = y$. Then $(\alpha, \beta') \in (\phi^*)_A$ and hence, for any $y \in \underline{Y}$, we have $(r_\phi(Y), y) \in (\phi^*)_A^{\downarrow Y}$.

Similarly, if $Y \in S_\phi$, all tuples in $(\alpha, \beta) \in \phi^*$ satisfy $\beta(S_\phi) = s_\phi$, and thus $y' = s_\phi(Y)$. For $(\alpha, \beta) \in (\phi^*)_A$ and any $y \in \underline{Y}$, we can define $\alpha'(X) = \alpha(X)$ for all $X \in \mathcal{V} \setminus \{Y\}$, and $\alpha'(Y) = y$. Then $(\alpha', \beta) \in (\phi^*)_A$ and hence, for any $y \in \underline{Y}$, we have $(y, s_\phi(Y)) \in (\phi^*)_A^{\downarrow Y}$.

Consider the case $Y \in W_\phi$, i.e., $Y \notin R_\phi \cup S_\phi \cup U_\phi \cup T_\phi \cup A$. Then, the tuples of extensions $(\alpha, \beta) \in \phi^*$ include all possible values for variable Y. Hence, $(\phi^*)_A^{\downarrow Y} = \underline{Y} \times \underline{Y}$. □

The following lemma will be used later.

Lemma 8.2. *Consider any $\phi \in \mathcal{L}_{pqT}$, and any set of variables $A \subseteq V$. We have the following.*

(i) There exists $(\alpha, \beta) \in \phi^*$ such that $\alpha(A) = \beta(A)$ if and only if $R_\phi \cap S_\phi \cap A = \emptyset$.

(ii) $\alpha(A) = \beta(A)$ holds for all $(\alpha, \beta) \in \phi^*$ if and only if $A \subseteq T_\phi \cup U_\phi$.

Proof. (i) First suppose that $R_\phi \cap S_\phi \cap A \neq \emptyset$, choose some $X \in R_\phi \cap S_\phi \cap A$, and consider any $(\alpha, \beta) \in \phi^*$. Then $\alpha(X) = r_\phi(X)$ and $\beta(X) = s_\phi(X) \neq \alpha(X)$, which shows that $\alpha(A) \neq \beta(A)$. Conversely, suppose that $R_\phi \cap S_\phi \cap A = \emptyset$. Let s'_ϕ be s_ϕ restricted to $S_\phi \setminus R_\phi$. Let α be any alternative extending u_ϕ and r_ϕ and s'_ϕ. Define β by $\beta(X) = s_\phi(X)$ if $X \in S_\phi$, and $\beta(X) = \alpha(X)$, otherwise. Then $(\alpha, \beta) \in \phi^*$ and $\alpha(A) = \beta(A)$, since α and β differ only on $R_\phi \cap S_\phi$, which is disjoint from A.

(ii) Assume first that $A \not\subseteq T_\phi \cup U_\phi$, i.e., $(R_\phi \cup S_\phi \cup W_\phi) \cap A \neq \emptyset$. We will construct α and β such that $\alpha(A) \neq \beta(A)$ and $(\alpha, \beta) \in \phi^*$. Let α be any alternative extending u_ϕ and r_ϕ and such that $\alpha(X) \neq s_\phi(X)$ for all $X \in S_\phi$. Let β be any alternative extending u_ϕ and s_ϕ and $\alpha(T_\phi)$ and such that $\beta(X) \neq r_\phi(X)$ for all $X \in R_\phi$, and also $\beta(X) \neq \alpha(X)$ for all $X \in W_\phi$ (we can do this because each element of the domain of each variable in $R_\phi \cup S_\phi \cup W_\phi$ includes at least two elements). Then $(\alpha, \beta) \in \phi^*$ and $\alpha(X) \neq \beta(X)$ for all $X \in R_\phi \cup S_\phi \cup W_\phi$, which implies that $\alpha(A) \neq \beta(A)$.

To prove the converse, assume that $A \subseteq T_\phi \cup U_\phi$ and consider any $(\alpha, \beta) \in \phi^*$. Then, for each $X \in T_\phi \cup U_\phi$, we have $\alpha(X) = \beta(X)$, and so $\alpha(A) = \beta(A)$. □

8.1.2 Checking Γ-Satisfaction

In this section, we investigate under what conditions a preference model satisfies preference statements Γ in $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$. Recall from Definition 3.7 that $\mathcal{L}'_{pqT}$ is $\mathcal{L}_{pqT}$ with certain negated statements also included. Formally, we define $\mathcal{L}'_{pqT}$ to be the union $\mathcal{L}_{pqT} \cup \{\neg\phi : \phi \in \mathcal{L}_{pqT}, \phi$ *non-strict, and* $R_\phi = S_\phi\}$

The next lemma shows a condition for variables in W_ϕ for $\phi^{(\succeq)}$ satisfying models.

Lemma 8.3. *Let $\phi \in \mathcal{L}_{pqT}$ and $\pi \in \mathscr{L}$. Suppose that $\pi \models \phi^{(\succeq)}$, i.e., $\succcurlyeq_\pi \supseteq \phi^*$. If $W_\phi \cap V_\pi \neq \emptyset$ then there exists $X \in R_\phi \cap S_\phi \cap V_\pi$ that appears earlier in π than any variable in W_ϕ.*

Proof. Suppose otherwise, and let X be the first variable in W_ϕ that appears in π, and let $\geq_X$ be the corresponding value ordering. We will define two different pairs (α, β) and (α', β') in ϕ^*. Let s'_ϕ be s_ϕ restricted to $S_\phi \setminus R_\phi$. Let α be any alternative extending u_ϕ and r_ϕ and s'_ϕ. Define β by: $\beta(X)$ is an element other than $\alpha(X)$; $\beta(Y) = s_\phi(Y)$ if $Y \in S_\phi$; $\beta(Y) = \alpha(Y)$ for all other Y. Then $(\alpha, \beta) \in \phi^*$, and α and β only differ on variable X and variables $R_\phi \cap S_\phi$. The first variable in π on which α and β differ is X, and thus, $\alpha(X) >_X \beta(X)$, since $\alpha \succcurlyeq_\pi \beta$.

Now, define alternative α' which agrees with α except on X, and alternative β' which agrees with β except on X, and where $\alpha'(X) = \beta(X)$ and $\beta'(X) = \alpha(X)$. By the same argument, we have $(\alpha', \beta') \in \phi^*$ and $\alpha'(X) >_X \beta'(X)$, i.e., $\beta(X) >_X \alpha(X)$, which is a contradiction, since $>_X$ is a total order. □

The following result characterises when a lex model satisfies a non-strict preference statement in $\mathcal{L}_{pqT}$.

Proposition 8.4. *Let $\pi \in \mathscr{L}$ and ϕ be a non-strict element of $\mathcal{L}_{pqT}$, so that $\phi = \phi^{(\succeq)}$. Let us say that $X \in V_\pi$ is* definite *if $X \in (R_\phi \cap S_\phi) \cup W_\phi$, and that X is* relevant *if $X \in R_\phi \cup S_\phi \cup W_\phi$ and there is no earlier definite variable in V_π. Thus, the set of relevant variables consists of the earliest definite variable (if there is one), plus all earlier variables not in T_ϕ or U_ϕ. As usual, we let $\geq_X$ be the total ordering associated with X in π. Then, $\pi \models \phi$ if and only if for all relevant variables X,*

(a) $X \notin W_\phi$;

(b) if $X \in R_\phi \cap S_\phi$ then $r_\phi(X) >_X s_\phi(X)$;

(c) if $X \in R_\phi \setminus S_\phi$ then for all $x \in \underline{X}$, $r_\phi(X) \geq_X x$, i.e., $r_\phi(X)$ is the best value of X; and

(d) if $X \in S_\phi \setminus R_\phi$ then for all $x \in \underline{X}$, $x \geq_X s_\phi(X)$, i.e., $s_\phi(X)$ is the worst value of X.

In particular, if $V_\pi \subseteq T_\phi \cup U_\phi$ then $\pi \models \phi$.

Proof. First let us assume that $\pi \models \phi$, i.e., $\pi \models \phi^{(\geq)}$, and thus, $\succcurlyeq_\pi \supseteq \phi^*$. Consider any relevant variable X. We will prove that (a), (b), (c) and (d) hold.

We first define a pair (α_0, β_0) in ϕ^*. Let s'_ϕ be s_ϕ restricted to $S_\phi \setminus R_\phi$. Let α_0 be any alternative extending u_ϕ and r_ϕ and s'_ϕ. Define β_0 by: $\beta_0(Y) = s_\phi(Y)$ if $Y \in S_\phi$; $\beta_0(Y) = \alpha_0(Y)$ for all other Y. The only variables on which α_0 and β_0 differ are those in $R_\phi \cap S_\phi$, and we have $\alpha_0(R_\phi) = r_\phi$ and $\beta_0(S_\phi) = s_\phi$. We thus have $(\alpha_0, \beta_0) \in \phi^*$.

(a): Suppose that $X \in W_\phi$. Let x be any element of $\underline{X}$ other than $\alpha_0(X)$, and let $x' = \alpha_0(X)$. Define β_1 by $\beta(X) = x$, and for all other $Y \in V \setminus \{X\}$, $\beta_1(Y) = \beta_0(Y)$. Also, define α_1 by $\alpha_1(X) = x$, and for all $Y \in V \setminus \{X\}$, $\alpha_1(Y) = \alpha_0(Y)$. It follows that (α_0, β_1) and (α_1, β_0) are in ϕ^*, and thus, $\alpha_0 \succcurlyeq_\pi \beta_1$ and $\alpha_1 \succcurlyeq_\pi \beta_0$, because $\succcurlyeq_\pi \supseteq \phi^*$. The first variable in π on which α_0 and β_1 differ is X, and thus, $\alpha_0(X) >_X \beta_1(X)$, i.e., $x' >_X x$. Similarly, the first variable in π on which α_1 and β_0 differ is X, and thus, $\alpha_1(X) >_X \beta_0(X)$, i.e., $x >_X x'$, contradicting the fact that $\geq_X$ is a total order.

(b): Assume that $X \in R_\phi \cap S_\phi$, and so $\alpha_0(X) \neq \beta_0(X)$. Since $(\alpha_0, \beta_0) \in \phi^*$ we have $\alpha_0 \succcurlyeq_\pi \beta_0$. Let Y be the first variable on which α_0 and β_0 differ, so $Y \in R_\phi \cap S_\phi$. Y is thus a definite variable. Since X is relevant, there is no earlier definite variable, so $Y = X$, and X is the first variable on which α_0 and β_0 differ. $\alpha_0 \succcurlyeq_\pi \beta_0$ implies that $\alpha_0(X) >_X \beta_0(X)$, i.e., $r_\phi(X) >_X s_\phi(X)$, proving (b).

(c): Assume that $X \in R_\phi \setminus S_\phi$ and $\pi \models \phi$. Choose any $x \in \underline{X}$ with $x \neq r_\phi(X)$. Let β_2 be an alternative that only differs with β_0 on X, and with $\beta_2(X) = x$. Then, $(\alpha_0, \beta_2) \in \phi^*$, and so $\alpha_0 \succcurlyeq_\pi \beta_2$, since $\pi \models \phi$. Now, α_0 and β_2 do not differ on any earlier variables, since no earlier variable is in $R_\phi \cap S_\phi$, because X is relevant. This implies that $\alpha_0(X) \geq_X \beta_2(X)$, i.e., $r_\phi(X) \geq_X x$.

(d): Assume that $X \in S_\phi \setminus R_\phi$ and $\pi \models \phi$. The proof of (d) is analogous to that of (c). Choose any $x \in \underline{X}$ with $x \neq s_\phi(X)$. Let α_2 be an alternative that

only differs with α_0 on X, and with $\alpha_2(X) = x$. Then, $(\alpha_2, \beta_0) \in \phi^*$, and so, $\alpha_2(X) \geq_X \beta_0(X)$, i.e., $x \geq_X s_\phi(X)$.

To prove the converse, we now assume that for all relevant variables, conditions (a), (b), (c) and (d) hold. We will prove that $\pi \models \phi$. It is sufficient to show that for all $(\alpha, \beta) \in \phi^*$ we have $\alpha \succcurlyeq_\pi \beta$. So, consider any $(\alpha, \beta) \in \phi^*$. If $\alpha(V_\pi) = \beta(V_\pi)$ then we have $\alpha \succcurlyeq_\pi \beta$, so we can assume that α and β differ on some variable in V_π; let X be the first such variable, and let A be the set of earlier variables, so that $\alpha(A) = \beta(A)$. By the definition of a cvo lexicographic order, to prove that $\alpha \succcurlyeq_\pi \beta$, it is sufficient to prove that $\alpha(X) \geq_X \beta(X)$ (i.e., $\alpha(X) >_X \beta(X)$, since $\alpha(X) \neq \beta(X)$).

We will show that X is relevant, by first showing that A contains no definite variable. Suppose that there exists a definite variable, and let Y be the earliest (according, as always, to the V_π ordering in π). Then Y is relevant. By condition (a), $Y \notin W_\phi$ and so $Y \in R_\phi \cap S_\phi$, but then $\alpha(Y) = r_\phi(Y) \neq s_\phi(Y) = \beta(Y)$, so $\alpha(Y) \neq \beta(Y)$. In particular this implies that $Y \notin A$, so A contains no definite variable. Since $\alpha(X) \neq \beta(X)$, we have $X \notin T_\phi \cup U_\phi$, so X is relevant.

If $X \in R_\phi \cap S_\phi$ then the definition of ϕ^* implies that $\alpha(X) = r_\phi(X)$ and $\beta(X) = s_\phi(X)$, and thus, $\alpha(X) >_X \beta(X)$, by condition (b). If $X \in R_\phi \setminus S_\phi$ then $\alpha(X) = r_\phi(X)$ and condition (c) implies that $\alpha(X) \geq_X \beta(X)$. Similarly, if $X \in S_\phi \setminus R_\phi$ then condition (d) implies that $\alpha(X) \geq_X \beta(X)$. □

We state the following corollary which is an immediate consequence of Proposition 8.4.

Corollary 8.5. *Suppose that $\phi \in \mathcal{L}_{pqT}$ and $\pi \in \mathscr{L}$ such that $\pi \models \phi$ and $V_\pi \cap R_\phi \cap S_\phi = \emptyset$. Then, $V_\pi \cap W_\phi = \emptyset$.*

Proof. If $\pi \models \phi$, then also $\pi \models \phi^{(\geq)}$. Suppose $V_\pi \cap W_\phi \neq \emptyset$. Let X be the first variable in $V_\pi \cap W_\phi$ that appears in the sequence of π. Since $V_\pi \cap R_\phi \cap S_\phi = \emptyset$, X is relevant. Then Proposition 8.4 a) implies that $X \notin W_\phi$, which is a contradiction. □

The next result gives the extra conditions required for satisfying strict statements.

Proposition 8.6. *Let $\phi \in \mathcal{L}_{pqT}$ and $\pi \in \mathscr{L}$.*

- *If ϕ is a fully strict statement, then $\pi \models \phi$ if and only if $\pi \models \phi^{(\geq)}$ and $R_\phi \cap S_\phi \cap V_\pi \neq \emptyset$.*

- *If ϕ is a weakly strict statement, then $\pi \models \phi$ if and only if $\pi \models \phi^{(\geq)}$ and $(R_\phi \cup S_\phi) \cap V_\pi \neq \emptyset$.*

Proof. The definitions immediately imply that if $\pi \models \phi$ then $\pi \models \phi^{(\geq)}$, so we can assume that in all cases $\pi \models \phi^{(\geq)}$.

Suppose that ϕ is a fully strict statement. Now $\pi \models \phi^{(\geq)}$ implies that $\alpha \succcurlyeq_\pi \beta$ for all $(\alpha, \beta) \in \phi^*$. Therefore, $\pi \models \phi$ if and only if for all $(\alpha, \beta) \in \phi^*$, $\alpha \not\equiv_\pi \beta$, i.e., $\alpha(V_\pi) \neq \beta(V_\pi)$. Lemma 8.2(i) then implies that $\pi \models \phi$ if and only if $R_\phi \cap S_\phi \cap V_\pi \neq \emptyset$.

Assume now that ϕ is a weakly strict statement, and also assume that $\pi \models \phi^{(\geq)}$. We then have $\pi \models \phi$ if and only if there exists $(\alpha, \beta) \in \phi^*$ with $\alpha(V_\pi) \neq \beta(V_\pi)$, which, by Lemma 8.2(ii), is if and only if $T_\phi \cup U_\phi \not\supseteq V_\pi$, i.e., $(R_\phi \cup S_\phi \cup W_\phi) \cap V_\pi \neq \emptyset$. Now, Corollary 8.5 implies that if $\pi \models \phi$ and $W_\phi \cap V_\pi \neq \emptyset$ then $R_\phi \cap V_\pi \neq \emptyset$, and thus, $\pi \models \phi$ if and only if $(R_\phi \cup S_\phi) \cap V_\pi \neq \emptyset$. □

Theorem 8.1: Γ-Satisfaction of $\models^*$-Model

Suppose that $\Gamma \subseteq \mathcal{L}'_{pqT}$ and that $\pi \models^* \Gamma$.

- If $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ and ϕ is non-strict, then $\pi \models \phi$.
- If $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ and ϕ is fully strict, then $\pi \models \phi \iff R_\phi \cap S_\phi \cap V_\pi \neq \emptyset$.
- If $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ and ϕ is weakly strict, then $\pi \models \phi \iff (R_\phi \cup S_\phi) \cap V_\pi \neq \emptyset$.
- If $\neg\phi \in \Gamma$, where ϕ is a non-strict element of $\mathcal{L}_{pqT}$ with $R_\phi = S_\phi$, we have $\pi \models \neg\phi \iff V_\pi \not\subseteq T_\phi \cup U_\phi$.

Thus, $\pi \models \Gamma$ if and only if

- for all fully strict statements ϕ in $\Gamma \cap \mathcal{L}_{pqT}$, $R_\phi \cap S_\phi \cap V_\pi \neq \emptyset$;
- for all weakly strict statements ϕ in $\Gamma \cap \mathcal{L}_{pqT}$, $(R_\phi \cup S_\phi) \cap V_\pi \neq \emptyset$;
- for all $\neg\phi \in \Gamma$, where ϕ is a non-strict element of $\mathcal{L}_{pqT}$ with $R_\phi = S_\phi$, we have $V_\pi \not\subseteq T_\phi \cup U_\phi$.

Proof. First consider any $\phi \in \Gamma \cap \mathcal{L}_{pqT}$. We have that $\pi \models^* \phi$, which implies, by Theorem 4.2, that $\pi \models \phi^{(\geq)}$. Thus, if ϕ is non-strict then $\pi \models \phi$, showing the first bullet point. We can then use Proposition 8.6 to imply the second and third bullet points.

Now consider an element of the form $\neg\phi$ in Γ. Theorem 4.2 implies that either $\pi \models \neg\phi$ or $V_\pi \cap S_\phi = \emptyset$. Thus, if $\pi \models \phi$ then $V_\pi \cap S_\phi = \emptyset$, and so $V_\pi \cap W_\phi = \emptyset$, by Corollary 8.5, and so $V_\pi \cap (R_\phi \cup S_\phi \cup W_\phi) = \emptyset$, i.e., $V_\pi \subseteq T_\phi \cup U_\phi$. Conversely, if $V_\pi \subseteq T_\phi \cup U_\phi$ then it follows using Proposition 8.4 that $\succcurlyeq_\pi \supseteq \phi^*$, and hence $\pi \models \phi^{(\geq)}$, and so, $\pi \models \phi$, since ϕ is a non-strict statement. This proves the fourth bullet point.

The second half of the result follows from the first half. □

8.1.3 $\models^*$-Models for Subsets of $\mathcal{L}'_{pqT}$

Theorem 4.2 suggests the feasibility of checking consistency of subsets of the language $\mathcal{L}'_{pqT}$.

We use the method of Section 4.2.3 to determine the consistency of a set of preference statements $\Gamma \subseteq \mathcal{L}'_{pqT}$, by incrementally extending a maximal $\models^*$-model π of Γ, and then checking whether or not $\pi \models \Gamma$ holds; this makes use of Theorem 8.1.

Definition 8.1: Best$^\pi_\Gamma(X)$, Worst$^\pi_\Gamma(X)$ and Pairs$^\pi_\Gamma(X)$

Let $\Gamma \subseteq \mathcal{L}'_{pqT}$, let $X \in \mathcal{V}$, and let $\pi \in \mathscr{L}$. Furthermore, let $\overline{\Gamma}$ be the set of all $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ such that $R_\phi \cap S_\phi \cap V_\pi = \emptyset$. We define:

- Best$^\pi_\Gamma(X) = \{r_\phi(X) : \phi \in \overline{\Gamma} \,\&\, X \in R_\phi \setminus S_\phi\}$.
- Worst$^\pi_\Gamma(X) = \{s_\phi(X) : \phi \in \overline{\Gamma} \,\&\, X \in S_\phi \setminus R_\phi\}$.
- Pairs$^\pi_\Gamma(X) =$ Pos$^\pi_\Gamma(X) \cup$ Neg$^\pi_\Gamma(X)$, where

Pos$^\pi_\Gamma(X)$ is the set of all pairs $(r_\phi(X), s_\phi(X))$ such that $\phi \in \overline{\Gamma}$ and $X \in R_\phi \cap S_\phi$. Neg$^\pi_\Gamma(X)$ is the set of all pairs $(s_\phi(X), r_\phi(X))$ such that $\neg\phi \in \Gamma$ and $T_\phi \cup U_\phi \supseteq V_\pi$, and $X \in R_\phi (= S_\phi)$.

Lemma 8.7. *Suppose that $\Gamma \subseteq \mathcal{L}'_{pqT}$. Let $X \in \mathcal{V}$ and let $\geq_X$ be a total ordering on $\underline{X}$, and let $\pi' = \pi \circ (X, \geq_X)$. Suppose that $\pi' \models^* \Gamma$. Then the following hold:*

- *For all $x \in$ Best$^\pi_\Gamma(X)$ and $x' \in \underline{X}$ we have $x \geq_X x'$. In particular then, $|$Best$^\pi_\Gamma(X)$$| \leq 1$.*
- *For all $x \in$ Worst$^\pi_\Gamma(X)$ and $x' \in \underline{X}$ we have $x' \geq_X x$. In particular then, $|$Worst$^\pi_\Gamma(X)$$| \leq 1$.*

- *If* $(x, x') \in \mathit{Pairs}^{\pi}_{\Gamma}(X)$ *then* $x \geq_X x'$.

Proof. Since $\pi' \models^* \Gamma$ we have, by Theorem 4.2, $\pi' \models \phi^{(\geq)}$ for $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, and for $\neg\phi \in \Gamma$, either $\pi' \models \neg\phi$ or $V_{\pi'} \cap S_\phi = \emptyset$. Recall the definitions of definite and relevant variables in Proposition 8.4. Given $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, we have that if ϕ is such that $R_\phi \cap S_\phi \cap V_\pi = \emptyset$ and $X \in R_\phi \cup S_\phi$ then X is relevant given $\phi^{(\geq)}$ and π'. This is because X would only not be relevant if there were an earlier definite variable Y in π' and thus in V_π; we'd then have $Y \notin W_\phi$, by Proposition 8.4 and so $Y \in R_\phi \cap S_\phi$, which contradicts $R_\phi \cap S_\phi \cap V_\pi = \emptyset$.

Suppose that $x \in \mathrm{Best}^{\pi}_{\Gamma}(X)$. By definition, there exists $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ such that $x = r_\phi(X)$ and $R_\phi \cap S_\phi \cap V_\pi = \emptyset$ and $X \in R_\phi \setminus S_\phi$. Since X is relevant given $\phi^{(\geq)}$ and π', Proposition 8.4 implies that for all $x' \in \underline{X}$, $x \geq_X x'$. Since $\geq_X$ is a total order, there can be at most one element x in $\mathrm{Best}^{\pi}_{\Gamma}(X)$. A similar argument shows that if $x \in \mathrm{Worst}^{\pi}_{\Gamma}(X)$ then for all $x' \in \underline{X}$, we have $x' \geq_X x$ which implies that $|\mathrm{Worst}^{\pi}_{\Gamma}(X)| \leq 1$.

Suppose that $(x, x') \in \mathrm{Pos}^{\pi}_{\Gamma}(X)$. Then, by definition, there exists $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ such that $r_\phi(X) = x$ and $s_\phi(X) = x'$ and $R_\phi \cap S_\phi \cap V_\pi = \emptyset$ and $X \in R_\phi \cap S_\phi$. Since X is relevant given $\phi^{(\geq)}$ and π', Proposition 8.4 implies that $x \geq_X x'$.

Suppose that $(x, x') \in \mathrm{Neg}^{\pi}_{\Gamma}(X)$. Then there exists $\neg\phi \in \Gamma$ with $s_\phi(X) = x$ and $r_\phi(X) = x'$ and $T_\phi \cup U_\phi \supseteq V_\pi$ and $X \in R_\phi = S_\phi$. Since $V_{\pi'} \cap S_\phi \neq \emptyset$ and $\pi' \models^* \neg\phi$, we have $\pi' \models \neg\phi$, by Theorem 4.2, i.e., $\pi' \not\models \phi$. The condition $T_\phi \cup U_\phi \supseteq V_\pi$, using Proposition 8.4 implies that $\pi \models \phi^{(\geq)}$, i.e., $\pi \models \phi$, since ϕ is non-strict. Also, X is relevant given $\phi^{(\geq)}$ and π', so, $\pi' \not\models \phi$ implies, using Proposition 8.4, that $r_\phi(X) \not\geq_X s_\phi(X))$, and thus, $s_\phi(X) >_X r_\phi(X)$ and $x \geq_X x'$. □

Definition 8.2: Variables That Can Be Chosen Next

Given $\Gamma \subseteq \mathcal{L}'_{pqT}$ and $\pi \in \mathscr{L}$ with $\pi \models^* \Gamma$, we say that X *can be chosen next* if: $X \in \mathcal{V} \setminus V_\pi$ and

- if $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ and $R_\phi \cap S_\phi \cap V_\pi = \emptyset$ then $X \notin W_\phi$;
- $\mathrm{Pairs}^{\pi}_{\Gamma}(X)$ is acyclic;
- $|\mathrm{Best}^{\pi}_{\Gamma}(X)| \leq 1$ and $|\mathrm{Worst}^{\pi}_{\Gamma}(X)| \leq 1$;
- if $x \in \mathrm{Best}^{\pi}_{\Gamma}(X)$ then x is undominated in $\mathrm{Pairs}^{\pi}_{\Gamma}(X)$, i.e., there exists no element of the form (x', x) in $\mathrm{Pairs}^{\pi}_{\Gamma}(X)$;

- if $x \in \text{Worst}^{\pi}_{\Gamma}(X)$ then x is not dominating in $\text{Pairs}^{\pi}_{\Gamma}(X)$, i.e., there exists no element of the form (x, x') in $\text{Pairs}^{\pi}_{\Gamma}(X)$.

Definition 8.3: Valid Extensions

Given $\Gamma \subseteq \mathcal{L}'_{pqT}$ and $\pi \in \mathscr{L}$ with $\pi \models^* \Gamma$, we say that $(X, \geq_X)$ is a *valid extension of* π if

(i) X can be chosen next,

(ii) $\geq_X \supseteq \text{Pairs}^{\pi}_{\Gamma}(X)$,

(iii) if $x \in \text{Best}^{\pi}_{\Gamma}(X)$, then x is the best element in $\underline{X}$ with respect to $\geq_X$ (so that $x \geq y$ for all $y \in \underline{X}$),

(iv) if $x' \in \text{Worst}^{\pi}_{\Gamma}(X)$ then x' is the worst element in $\underline{X}$ with respect to $\geq_X$.

Note that, for any variable X that can be chosen next, there exists a valid extension $(X, \geq_X)$.

The following result states the conditions needed for minimally extending π to maintain the $\models^*$-satisfaction of Γ.

Proposition 8.8. *Suppose that $\Gamma \subseteq \mathcal{L}'_{pqT}$, and that $\pi \models^* \Gamma$. Let X be a variable in $\mathcal{V} \setminus V_\pi$ and let $\pi' = \pi \circ (X, \geq_X)$, where $\geq_X$ is a total ordering on $\underline{X}$. Then $\pi' \models^* \Gamma$ if and only if $(X, \geq_X)$ is a valid extension of π.*

Proof. Since $\pi \models^* \Gamma$, Theorem 4.2 implies that $\pi \models \phi^{(\geq)}$ for $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, and for $\neg\phi \in \Gamma$, either $\pi \models \neg\phi$ or $V_\pi \cap S_\phi = \emptyset$ (since $\neg\phi \in \mathcal{L}'_{pqT}$ implies that $\phi \in \mathcal{L}_{pqT}$ and ϕ is non-strict, and $R_\phi = S_\phi$).

$\Leftarrow$: We will first prove that if $(X, \geq_X)$ is a valid extension of π then $\pi' \models^* \Gamma$. For $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, we have $\pi' \models^* \phi$ if and only if $\pi' \models \phi^{(\geq)}$, by Theorem 4.2. Also, for $\neg\phi \in \Gamma$ we have $\pi' \models^* \neg\phi$ if and only if either $\pi' \models \neg\phi$ or $V_{\pi'} \cap S_\phi = \emptyset$.

Consider any $\phi \in \Gamma \cap \mathcal{L}_{pqT}$. Since $\pi \models \phi^{(\geq)}$, it follows using Lemma 4.44 that $\pi' \models \phi^{(\geq)}$ if and only if $\geq_X \supseteq \phi^{\downarrow X}_{V_\pi}$.

Consider any $(x, x') \in \phi^{\downarrow X}_{V_\pi}$. We need to show that $(x, x') \in \geq_X$, i.e., that $x \geq_X x'$; this will then imply that $\geq_X \supseteq \phi^{\downarrow X}_{V_\pi}$, and hence, $\pi' \models \phi^{(\geq)}$. Since $\phi^{\downarrow X}_{V_\pi}$ is non-empty, we have, using Proposition 8.1, that $R_\phi \cap S_\phi \cap V_\pi = \emptyset$. This implies

that $X \notin W_\phi$, because $(X, \geq_X)$ is a valid extension of π. If $x = x'$ then clearly, $x \geq_X x'$, This covers the cases when $X \in T_\phi$ and $X \in U_\phi$ (see Proposition 8.1).

If $X \in R_\phi \cap S_\phi$, then, by Proposition 8.1, $\phi^{\downarrow X}_{V_\pi} = \{(r_\phi(X), s_\phi(X))\}$. Thus, $(x, x') \in \mathrm{Pos}^\pi_\Gamma(X)$, so $x \geq_X x'$ since $(X, \geq_x)$ is a valid extension of π.

If $X \in R_\phi \setminus S_\phi$ then $r_\phi(X) \in \mathrm{Best}^\pi_\Gamma(X)$. Proposition 8.1 implies that $x = r_\phi(X)$, and thus, $x \geq_X x'$. Similarly, if $X \in S_\phi \setminus R_\phi$ then $s_\phi(X) \in \mathrm{Worst}^\pi_\Gamma(X)$. Proposition 8.1 implies that $x' = s_\phi(X)$, and thus, $x \geq_X x'$. This completes the proof that, for any $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, we have $\geq_X \supseteq \phi^{\downarrow X}_{V_\pi}$, and hence, $\pi' \models \phi^{(\geq)}$, and thus, $\pi' \models^* \phi$.

Now suppose that $\neg\phi \in \Gamma$, and so ϕ is non-strict and $R_\phi = S_\phi$. We will show that $\pi' \models^* \neg\phi$. Since $\pi \models^* \neg\phi$ we have either $\pi \models \neg\phi$ or $V_\pi \cap S_\phi = \emptyset$. If $\pi \models \neg\phi$, and so $\succcurlyeq_\pi \not\supseteq \phi^*$, then the fact that π' extends π implies that $\succcurlyeq_{\pi'} \subseteq \succcurlyeq_\pi$ (e.g., using Lemma 4.33), and thus, $\succcurlyeq_{\pi'} \not\supseteq \phi^*$, and therefore $\pi' \models \neg\phi$ and $\pi' \models^* \neg\phi$. We now thus have only to consider the case when $\pi \models \phi$ and $V_\pi \cap S_\phi = \emptyset$. This implies, using Corollary 8.5, that $V_\pi \cap W_\phi = \emptyset$, and thus, $V_\pi \subseteq T_\phi \cup U_\phi$. If $X \notin S_\phi$ then $V_{\pi'} \cap S_\phi = \emptyset$, and so, $\pi' \models^* \neg\phi$. Now assume that $X \in S_\phi$. This implies that $(s_\phi(X), r_\phi(X)) \in \mathrm{Neg}^\pi_\Gamma(X)$. Because $(X, \geq_X)$ is a valid extension of π, we have $s_\phi(X) \geq_X r_\phi(X)$, i.e., $s_\phi(X) >_X r_\phi(X)$, since $s_\phi(X) \neq r_\phi(X)$. It cannot be the case that $\pi' \models \phi$, since then we would have $\pi' \models \phi^{(\geq)}$ and thus, using Lemma 4.44, $\geq_X \supseteq \phi^{\downarrow X}_{V_\pi}$, which implies $r_\phi(X) \geq_X s_\phi(X)$ using Proposition 8.1, contradicting $s_\phi(X) >_X r_\phi(X)$. We therefore have $\pi' \models \neg\phi$, and thus, $\pi' \models^* \neg\phi$.

$\Rightarrow$: Assume now that $\pi' \models^* \Gamma$; we will show that $(X, \geq_X)$ is a valid extension of π. We have $X \in \mathcal{V} \setminus V_\pi$. Since $\pi' \models^* \Gamma$ we have $\pi' \models \phi^{(\geq)}$ for $\phi \in \Gamma \cap \mathcal{L}_{pqT}$, and for $\neg\phi \in \Gamma$, either $\pi' \models \neg\phi$ or $V_{\pi'} \cap S_\phi = \emptyset$. For $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ we then have $\geq_X \supseteq \phi^{\downarrow X}_{V_\pi}$, by Lemma 4.44. Corollary 8.5 implies that if $\phi \in \Gamma \cap \mathcal{L}_{pqT}$ and $R_\phi \cap S_\phi \cap V_\pi = \emptyset$ then $X \notin W_\phi$. Lemma 8.7 implies that there is at most one element in $\mathrm{Best}^\pi_\Gamma(X)$ and at most one element in $\mathrm{Worst}^\pi_\Gamma(X)$. Also if $(x, x') \in \mathrm{Pairs}^\pi_\Gamma(X)$ then $x \geq_X x'$ and so $\geq_X$ extends $\mathrm{Pairs}^\pi_\Gamma(X)$, and thus, $\mathrm{Pairs}^\pi_\Gamma(X)$ is acyclic. The same lemma also implies that if $x \in \mathrm{Best}^\pi_\Gamma(X)$ then for all $x' \in \underline{X}$, $x \geq_X x'$, and thus, by the acyclicity of $\geq_X$, x is undominated in $\mathrm{Pairs}^\pi_\Gamma(X)$. A similar argument shows that if $x \in \mathrm{Worst}^\pi_\Gamma(X)$ then x is not dominating in $\mathrm{Pairs}^\pi_\Gamma(X)$. This completes the proof that $(X, \geq_X)$ is a valid extension of π. □

8.1.4 The Algorithm

Based on Proposition 8.8 and Theorem 8.1, we give the following formal description of the algorithm to solve $\mathscr{L}$-Consistency for Statements $\Gamma \subseteq \mathcal{L}'_{pqT}$.

Algorithm 8.1: $\mathscr{L}$-Consistency for Statements $\Gamma \subseteq \mathcal{L}'_{pqT}$

```
1. π ← ()
2. WHILE ( ∃ X ∈ 𝒱 \ σ(π) that can be chosen next)
3.         Choose a valid extension (X, ≥_X) for such X
4.         π ← π ∘ (X, ≥_X)
5. FOR ( φ ∈ Γ ) DO
6.   IF(φ is fully strict and R_φ ∩ S_φ ∩ V_π = ∅, or
7.      φ is weakly strict and (R_φ ∪ S_φ) ∩ V_π = ∅, or
8.      φ is a negated statement and V_π ⊆ T_φ ∪ U_φ) THEN
9.      RETURN "Γ is inconsistent" and STOP.
10. RETURN "Γ is consistent" and STOP.
```

In summary, when building up a maximal $\models^*$-model π of Γ incrementally, at each stage we see if there is a variable X that can be chosen next. If so, we generate a valid extension; if not, we then have generated a maximal $\models^*$-model π of Γ (by Proposition 8.8). We check consistency of Γ by determining if π satisfies Γ, following the results of Theorem 8.1. Hence we use the general algorithm presented in Section 4.2.4, however, with detailed description on how to find the minimal extensions/variables that can be chosen next, and how ($\models^*$-)satisfaction tests can be executed.

Using the fact that $|\text{Pairs}^{\pi}_{\Gamma}(X)| \leq |\Gamma|$, it can be shown that the overall complexity of checking consistency for $\Gamma \subseteq \mathcal{L}'_{pqT}$ is $O(|\mathcal{V}|^2|\Gamma|)$, if variable domains are of constant size. The first for-loop explores all variables. Within the for-loop, we test if there exists a variable that can be chosen next. To check if a variable can be chosen next, we need to analyse the constraints on the value order of the considered variable given by the preference statements. This takes $O(|\text{Pairs}^{\pi}_{\Gamma}(X)|)$ time, which is $O(|\Gamma|)$. If a variable can be chosen next, we can construct a valid extension in the same time bound, since the variable domains are assumed to be of constant size. This gives us, for the first for-loop, a time of $O(\sum_{i=1,\ldots,|\mathcal{V}|} \sum_{j=1,\ldots,|\mathcal{V}|-i} |\Gamma|)$, which is $O(|\mathcal{V}|^2|\Gamma|)$. The second for-loop performs

satisfaction tests for all preference statements for the constructed $\models^*$-model. A satisfaction test for one preference statement can be done in $O(|\mathcal{V}|)$. Thus the time for the second for-loop is $O(|\mathcal{V}||\Gamma|)$. Hence, the overall running time is bound by $O(|\mathcal{V}|^2|\Gamma|)$.

Note that, to check consistency, we do not need to construct a satisfying model with specified value orders on the variable domains. It is enough to keep track of the relations of values that are constraint by the preference statements, i.e., the sets $\text{Best}^\pi_\Gamma(X)$, $\text{Worst}^\pi_\Gamma(X)$ and $\text{Pairs}^\pi_\Gamma(X)$. Their size is bound by $O(|\Gamma|)$, and so even for non-constant variable domains, the algorithm can be modified to run in $O(|\mathcal{V}|^2|\Gamma|)$.

In comparison, Section 6.3.3 described this procedure for fvo lexicographic models $\mathcal{H}(1)$ and less general preference statements $\mathcal{L}^{\mathcal{A}}$. In this case, finding minimal extensions is easier, and the overall complexity of the method was reduced by a factor of $|\mathcal{V}|$.

Example 8.1

Consider the set of cvo lexicographic models $\mathscr{L}$ over variables $\mathcal{V}$ = {airline, class,time} with domains orders {KLM, LAN}, {business, economy} and {day, night}. Let $\Gamma = \{\varphi_1, \varphi_2, \varphi_3, \varphi_4\} \subseteq \mathcal{L}'_{pqT}$ with:

φ_1: (LAN, business) $\geq$ (KLM, economy, night) | $\emptyset$
φ_2: (KLM) $>$ (economy, night) | $\emptyset$
φ_3: (KLM, day) $\gg$ (night) | $\emptyset$
φ_4: $\neg$ (KLM, economy) $\geq$ (LAN, business) | $\emptyset$

To find out if Γ is consistent, we start with the minimal model $\pi = ()$ and search for a variable that can be chosen next:
For variable airline, Best^π_Γ(airline)= {KLM}, but Pairs^π_Γ(airline)= {(LAN,KLM)}, i.e., KLM $\in$ Best^π_Γ(airline) is dominated by LAN in Pairs^π_Γ(airline). Thus, airline cannot be chosen next. Also, class cannot be chosen next as class $\in W_{\varphi_3}$.
For variable time, Pairs^π_Γ(time)= {(day,night)} is acyclic, and time $\notin W_{\varphi_1}$, time $\notin W_{\varphi_2}$ and time $\notin W_{\varphi_3}$. Also, $|\text{Best}^\pi_\Gamma(\text{time})| = |\{\}| \leq 1$ and $|\text{Worst}^\pi_\Gamma(\text{time})| = |\{\text{night}\}| \leq 1$ where night is not dominating in any element in Pairs^π_Γ(time). Thus, time can be chosen next. The tuple (time, day > night) is a valid extension and so we set π = ((time, day > night)).

Since the sets $\text{Best}^{\pi}_{\Gamma}$(airline) and $\text{Pairs}^{\pi}_{\Gamma}$(airline) remain unchanged, airline cannot be chosen next again. For variable class, $\text{Pairs}^{\pi}_{\Gamma}$(class)$=$ {(business,economy)} is acyclic, and class $\notin W_{\varphi_1}$, class $\notin W_{\varphi_2}$. (Note that W_{φ_3} is ignored at this point since $R_{\varphi_3} \cap S_{\varphi_3} \cap V_{\pi} \neq \emptyset$.) Also, $|\text{Best}^{\pi}_{\Gamma}(\text{class})| = |\{\}| \leq 1$ and $|\text{Worst}^{\pi}_{\Gamma}(\text{class})| = |\{\text{economy}\}| \leq 1$ where economy is not dominating in any element in $\text{Pairs}^{\pi}_{\Gamma}$(class). Thus, class can be chosen next. Then the tuple (class, business > economy) is a valid extension and so we set $\pi =$ ((time, day > night), (class, business > economy)).

Now, we can see that airline can be chosen next since $R_{\varphi_1} \cap S_{\varphi_1} \cap V_{\pi} \neq \emptyset$ and $R_{\varphi_3} \cap S_{\varphi_3} \cap V_{\pi} \neq \emptyset$, and thus $\text{Pairs}^{\pi}_{\Gamma}$(airline)$=$ {} is acyclic. Also, airline $\notin W_{\varphi_2}$, and $|\text{Best}^{\pi}_{\Gamma}(\text{airline})| = |\{\text{KLM}\}| \leq 1$ and $|\text{Worst}^{\pi}_{\Gamma}(\text{airline})| = |\{\}| \leq 1$ where KLM is not dominated in any element in $\text{Pairs}^{\pi}_{\Gamma}$(class). The tuple (airline, KLM > LAN) is a valid extension and so we set $\pi =$ ((time, day > night), (class, business > economy), (airline, KLM > LAN)).

Since there are no more variables left to add, π is a maximal $\models^*$-model of Γ. In fact, $\pi \models \Gamma$ and thus Γ is consistent.

8.2 Optimal Alternatives

Let us consider a finite set of alternatives $\mathcal{A}$, and assume that we have elicited a set Γ of preference statements from the user; we would like to find the optimal alternatives among $\mathcal{A}$ based on the user's preferences. As we will see in this section, there are several natural definitions of *optimal* [GPR$^+$10, WO11]. We compare some of these notions of optimality for cvo lexicographic models $\mathscr{L}$ in the next section and analyse their computational cost in Section 8.2.3.

8.2.1 Notions of Optimality

The definitions of different notions of optimality in this section are based on inferences $\models_{\mathscr{L}}$ for cvo lexicographic models $\mathscr{L}$. But can similarly be defined for other preference model types. Let Γ be a set of preference statements over some language $\mathcal{L}$.

Γ-Induced Order Relation We define the pre-order relation $\succcurlyeq_\Gamma$ on outcomes by $\alpha \succcurlyeq_\Gamma \beta \iff \Gamma \models_{\mathscr{L}} \alpha \geq \beta$. For $\alpha \succcurlyeq_\Gamma \beta$, we say that α *dominates* β. We define equivalence relation $\equiv_\Gamma$ by $\alpha \equiv_\Gamma \beta \iff \Gamma \models_{\mathscr{L}} \alpha \equiv \beta$, i.e., if α and β are equivalent in all models of Γ. For a set of alternatives B, we say that B are all Γ-equivalent if for all $\alpha, \beta \in B$, we have $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$. We also define $\succ_\Gamma$ to be the strict part of $\succcurlyeq_\Gamma$, so that $\alpha \succ_\Gamma \beta$ if and only if $\alpha \succcurlyeq_\Gamma \beta$ and $\alpha \not\equiv_\Gamma \beta$. We then say that α *strictly dominates* β.

Can Strictly Dominate We define $\mathrm{CSD}_\Gamma(\mathcal{A})$ ('Can Strictly Dominate') to be the set of maximal, i.e., undominated, elements of $\mathcal{A}$ w.r.t. $\succ_\Gamma$. $\alpha \in \mathrm{CSD}_\Gamma(\mathcal{A})$ if and only if for all $\beta \in \mathcal{A}$ which are not $\equiv_\Gamma$-equivalent to α there exists some $\pi \in \mathscr{L}$ with $\pi \models \Gamma$ and $\alpha \succ_\pi \beta$.

Necessarily Optimal We define $\mathrm{O}_\pi(\mathcal{A})$ to be the subset of the alternatives that are optimal in model $\pi \in \mathscr{L}$, i.e., $\{\alpha \in \mathcal{A} : \forall \beta \in \mathcal{A}, \alpha \succcurlyeq_\pi \beta\}$. We say that $\alpha \in \mathcal{A}$ is *necessarily optimal in* $\mathcal{A}$, written $\alpha \in \mathrm{NO}_\Gamma(\mathcal{A})$, if α is optimal in every model, i.e., if for all $\pi \in \mathscr{L}$ with $\pi \models \Gamma$ we have $\alpha \in \mathrm{O}_\pi(\mathcal{A})$. This holds if and only if for all $\beta \in \mathcal{A}$, and for all $\pi \in \mathscr{L}$ with $\pi \models \Gamma$ we have $\alpha \succcurlyeq_\Gamma \beta$.

Possibly (Strictly) Optimal We say that α is *possibly optimal*, written $\alpha \in \mathrm{PO}_\Gamma(\mathcal{A})$, if α is optimal in some model of Γ, so that $\mathrm{PO}_\Gamma(\mathcal{A}) = \bigcup_{\pi \models \Gamma} \mathrm{O}_\pi(A)$. Similarly, we say that $\alpha \in \mathrm{POM}_\Gamma(\mathcal{A})$ if α is optimal in some maximal model of Γ. Thus we have $\mathrm{POM}_\Gamma(\mathcal{A}) = \bigcup_{\pi \models^{max} \Gamma} \mathrm{O}_\pi(A)$, where $\pi \models^{max} \Gamma$ means that $\pi \in \mathscr{L}$ is a maximal model of Γ. α is *possibly strictly optimal* in $\mathcal{A}$, written $\alpha \in \mathrm{PSO}_\Gamma(\mathcal{A})$, if there exists some $\pi \in \mathscr{L}$ with $\pi \models \Gamma$ and $\mathrm{O}_\pi(A) \ni \alpha$ and $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$ for all $\beta \in \mathrm{O}_\pi(A)$. Thus α is in $\mathrm{PSO}_\Gamma(\mathcal{A})$ if there is a model of Γ in which α is optimal, and all other optimal elements are equivalent to α.

Strictly Optimal Given $\Gamma \subseteq \mathcal{L}$, we say that α is *strictly optimal* (within $\mathcal{A}$) with respect to $\pi \in \mathscr{L}$ if α is optimal in π and any other optimal element is equivalent to α, i.e., $\alpha \in \mathrm{O}_\pi(\mathcal{A})$ and $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$ for all $\beta \in \mathrm{O}_\pi(\mathcal{A})$. We write $\mathrm{SO}^\Gamma_\pi(\mathcal{A})$ for the set of such elements. If $\mathrm{O}_\pi(\mathcal{A})$ are all Γ-equivalent then $\mathrm{SO}^\Gamma_\pi(\mathcal{A}) = \mathrm{O}_\pi(\mathcal{A})$, otherwise, $\mathrm{SO}^\Gamma_\pi(\mathcal{A}) = \emptyset$. We always have that $\mathrm{SO}^\Gamma_\pi(\mathcal{A})$ are all Γ-equivalent.

Maximally Possibly Optimal Let $\text{Opt}_\Gamma^{\mathcal{A}}(\alpha)$ be the set of models $\pi \in \mathscr{L}$ of Γ that make α optimal in $\mathcal{A}$, i.e., $\text{Opt}_\Gamma^{\mathcal{A}}(\alpha) = \{\pi \in \mathscr{L} \mid \pi \models \Gamma, \text{O}_\pi(\mathcal{A}) \ni \alpha\}$. We define $\alpha \in \text{MPO}_\Gamma(\mathcal{A})$ if $\text{Opt}_\Gamma^{\mathcal{A}}(\alpha)$ is maximal, in the sense that there exists no $\beta \in \mathcal{A}$ with $\text{Opt}_\Gamma^{\mathcal{A}}(\beta)$ a strict superset of $\text{Opt}_\Gamma^{\mathcal{A}}(\alpha)$. We say that $\alpha \in \text{MPO}_\Gamma(\mathcal{A})$ is *maximally possibly optimal* in $\mathcal{A}$ given Γ; this holds if and only if there is no alternative that is optimal in the same set of cvo lexicographic models and more.

Extreme Elements Let $\pi_1, \ldots, \pi_k$ be a finite sequence of models. Define $\mathcal{A}_{\pi_1}$ to be $\text{O}_{\pi_1}(\mathcal{A})$. For $i = 1, \ldots, k$ we iteratively define $\mathcal{A}_{\pi_1,\ldots,\pi_i}$ to be $\text{O}_{\pi_i}(\mathcal{A}_{\pi_1,\ldots,\pi_{i-1}})$. We define the *extreme elements* $\text{EXT}_\Gamma(\mathcal{A})$ as follows. $\alpha \in \text{EXT}_\Gamma(\mathcal{A})$ if and only if there exists a sequence $\pi_1, \ldots, \pi_k$ of models of Γ such that $\mathcal{A}_{\pi_1,\ldots,\pi_k} \ni \alpha$ and for all $\beta \in \mathcal{A}_{\pi_1,\ldots,\pi_k}$, $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$. Therefore, $\alpha \in \text{EXT}_\Gamma(\mathcal{A})$ if there is a sequence of models such that iteratively maximising with respect to each model in turn leads to a set containing α and only other alternatives that are Γ-equivalent to α.

8.2.2 Optimality for $\mathscr{L}$ and Compositional Statements

In the following, we analyse and compare the different notions of optimality defined in the previous section for cvo lexicographic models $\mathscr{L}$ and compositional statements. We start by giving some basic properties.

For $\alpha \in \mathcal{A}$, let $\Delta_\alpha^{\mathcal{A}} = \{\alpha \geq \beta : \beta \in \mathcal{A}\}$.

Lemma 8.9. *Let $\Gamma \subseteq \mathcal{L}$ be a set of preference statements, and let π and π' be* cvo *lexicographic models, and let α be an element of set of alternatives $\mathcal{A}$. Then the following all hold.*

1. *If π' extends π then $\text{O}_{\pi'}(\mathcal{A}) \subseteq \text{O}_\pi(\mathcal{A})$.*
2. $\alpha \in \text{O}_\pi(\mathcal{A}) \iff \pi \models \Delta_\alpha^{\mathcal{A}}$
3. $\pi \models \Gamma \cup \Delta_\alpha^{\mathcal{A}} \iff \pi \in \textit{Opt}_\Gamma^{\mathcal{A}}(\alpha)$
4. $\alpha \in \text{PO}_\Gamma(\mathcal{A}) \iff \Gamma \cup \Delta_\alpha^{\mathcal{A}}$ *is consistent* $\iff \textit{Opt}_\Gamma^{\mathcal{A}}(\alpha)$ *is non-empty.*
5. $\Gamma \cup \Delta_\alpha^{\mathcal{A}} \models_{\mathscr{L}} \alpha \equiv \beta \iff \textit{Opt}_\Gamma^{\mathcal{A}}(\beta) \supseteq \textit{Opt}_\Gamma^{\mathcal{A}}(\alpha)$.

Proof. 1) Assume that π' extends π. Consider any $\alpha \in \text{O}_{\pi'}(\mathcal{A})$, so that, for all $\beta \in \mathcal{A}$, $\alpha \succcurlyeq_{\pi'} \beta$. By Lemma 4.34, if $\alpha \succcurlyeq_{\pi'} \beta$ then $\alpha \succcurlyeq_\pi \beta$. Therefore, for all

$\beta \in \mathcal{A}$, $\alpha \succcurlyeq_\pi \beta$, and so $\alpha \in \mathrm{O}_\pi(\mathcal{A})$.

2) We have: $\pi \models \Delta^{\mathcal{A}}_\alpha$ if and only if for all $\beta \in \mathcal{A}$, $\alpha \succcurlyeq_\pi \beta$, which is if and only if $\alpha \in \mathrm{O}_\pi(\mathcal{A})$.

3) $\pi \models \Gamma \cup \Delta^{\mathcal{A}}_\alpha$ if and only if $\pi \models \Gamma$ and $\alpha \in \mathrm{O}_\pi(\mathcal{A})$, which is if and only if $\pi \in \mathrm{Opt}^{\mathcal{A}}_\Gamma(\alpha)$.

4) $\alpha \in \mathrm{PO}_\Gamma(\mathcal{A})$ if and only if there exists some π with $\pi \models \Gamma$ and $\alpha \in \mathrm{O}_\pi(\mathcal{A})$. By 2), this holds if and only if there exists π with $\pi \models \Gamma \cup \Delta^{\mathcal{A}}_\alpha$, i.e., $\Gamma \cup \Delta^{\mathcal{A}}_\alpha$ is consistent. This is also equivalent to $\mathrm{Opt}^{\mathcal{A}}_\Gamma(\alpha)$ being non-empty by 3).

5) First suppose that $\Gamma \cup \Delta^{\mathcal{A}}_\alpha \models_{\mathscr{L}} \alpha \equiv \beta$, and consider any $\pi \in \mathrm{Opt}^{\mathcal{A}}_\Gamma(\alpha)$. Then, $\pi \models \Gamma \cup \Delta^{\mathcal{A}}_\alpha$, and thus, $\pi \models \alpha \equiv \beta$, and so, $\alpha \equiv_\pi \beta$. This implies that $\beta \in \mathrm{O}_\pi(\mathcal{A})$ and hence, $\pi \in \mathrm{Opt}^{\mathcal{A}}_\Gamma(\beta)$.
Conversely, suppose that $\mathrm{Opt}^{\mathcal{A}}_\Gamma(\beta) \supseteq \mathrm{Opt}^{\mathcal{A}}_\Gamma(\alpha)$, and consider any π such that $\pi \models \Gamma \cup \Delta^{\mathcal{A}}_\alpha$; we then have $\alpha \succcurlyeq_\pi \beta$. Then, by 3), $\pi \in \mathrm{Opt}^{\mathcal{A}}_\Gamma(\alpha)$, and so, $\pi \in \mathrm{Opt}^{\mathcal{A}}_\Gamma(\beta)$, which implies that $\pi \models \Gamma \cup \Delta^{\mathcal{A}}_\beta$. This entails that $\beta \succcurlyeq_\pi \alpha$, and thus $\alpha \equiv_\pi \beta$, i.e., $\pi \models \alpha \equiv \beta$. We have shown that $\Gamma \cup \Delta^{\mathcal{A}}_\alpha \models_{\mathscr{L}} \alpha \equiv \beta$. □

Without making assumptions about Γ we have the following properties from [WO11], which follow from basic arguments, that apply in a very general context (for proofs see also [O'M13]).

Proposition 8.10. *Consider a set of alternatives $\mathcal{A}$ and preference statements $\Gamma \subseteq \mathcal{L}$. Then, the following all hold. (i)* $\mathrm{NO}_\Gamma(\mathcal{A}) \cup \mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{MPO}_\Gamma(\mathcal{A}) \cap \mathrm{EXT}_\Gamma(\mathcal{A})$*; (ii)* $\mathrm{EXT}_\Gamma(\mathcal{A}) \subseteq \mathrm{CSD}_\Gamma(\mathcal{A}) \cap \mathrm{PO}_\Gamma(\mathcal{A})$*; (iii)* $\mathrm{MPO}_\Gamma(\mathcal{A}) \subseteq \mathrm{PO}_\Gamma(\mathcal{A})$*; (iv)* $\mathrm{MPO}_\Gamma(\mathcal{A}) \cap \mathrm{EXT}_\Gamma(\mathcal{A})$ *is always non-empty. (v) If* $\mathrm{NO}_\Gamma(\mathcal{A})$ *is non-empty then* $\mathrm{NO}_\Gamma(\mathcal{A}) = \mathrm{MPO}_\Gamma(\mathcal{A}) = \mathrm{EXT}_\Gamma(\mathcal{A}) = \mathrm{CSD}_\Gamma(\mathcal{A})$.

Proof. (i) Proposition 4.7 in [O'M13] implies that $\mathrm{NO}_\Gamma(\mathcal{A}) \cup \mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{EXT}_\Gamma(\mathcal{A})$. Furthermore, Proposition 4.6 in [O'M13] implies that $\mathrm{NO}_\Gamma(\mathcal{A}) \cup \mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{MPO}_\Gamma(\mathcal{A})$.

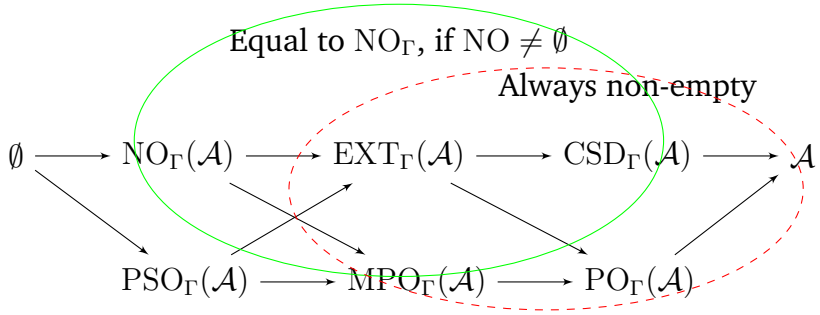
(ii) Proposition 4.7 in [O'M13] implies that $\mathrm{EXT}_\Gamma(\mathcal{A}) \subseteq \mathrm{CSD}_\Gamma(\mathcal{A})$ and $\mathrm{EXT}_\Gamma(\mathcal{A}) \subseteq \mathrm{PO}_\Gamma(\mathcal{A})$.

(iii) Proposition 4.6 in [O'M13] implies that $\mathrm{MPO}_\Gamma(\mathcal{A}) \subseteq \mathrm{PO}_\Gamma(\mathcal{A})$.

(iv) Proposition 4.7 in [O'M13] implies that $\mathrm{EXT}_\Gamma(\mathcal{A}) \cap \mathrm{MPO}_\Gamma(\mathcal{A})$ is always non-empty.

(v) Proposition 4.8 in [O'M13] implies that if $\mathrm{NO}_\Gamma(\mathcal{A})$ is non-empty then $\mathrm{NO}_\Gamma(\mathcal{A}) = \mathrm{MPO}_\Gamma(\mathcal{A}) = \mathrm{EXT}_\Gamma(\mathcal{A}) = \mathrm{CSD}_\Gamma(\mathcal{A})$. □

We can visualise these relations in the following diagram, where $A \to B$ represents the relation $A \subseteq B$.



The following lemmas and propositions will extend these results by relations of $\mathrm{POM}_\Gamma(\mathcal{A})$ and the case where Γ is compositional.

Lemma 8.11. *For alternatives $\mathcal{A}$ and preference statements $\Gamma \in \mathcal{L}$ under* cvo *lexicographic models, we have $\mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{POM}_\Gamma(\mathcal{A})$. If Γ is compositional, then $\mathrm{PSO}_\Gamma(\mathcal{A}) = \mathrm{POM}_\Gamma(\mathcal{A})$.*

Proof. Suppose that $\alpha \in \mathrm{PSO}_\Gamma(\mathcal{A})$, so there exists $\pi \in \mathscr{L}$ with $\pi \models \Gamma$, and $\mathrm{O}_\pi(\mathcal{A}) \ni \alpha$, and $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$ for all $\beta \in \mathrm{O}_\pi(\mathcal{A})$. Let π' be any maximal model of Γ that extends π. Choose some β that is optimal in π', i.e., $\beta \in \mathrm{O}_{\pi'}(\mathcal{A})$. Then, using Lemma 8.9, $\beta \in \mathrm{O}_\pi(\mathcal{A})$, and thus, $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$, so $\pi' \models \alpha \equiv \beta$, which implies that $\alpha \in \mathrm{O}_{\pi'}(\mathcal{A})$, and thus, $\alpha \in \mathrm{POM}_\Gamma(\mathcal{A})$.

Assume now that Γ is compositional. Let $\alpha \in \mathrm{POM}_\Gamma(\mathcal{A})$, so there exists $\pi \in \mathscr{L}$ with $\pi \models^{max} \Gamma$ and $\alpha \in \mathrm{O}_\pi(\mathcal{A})$. Proposition 4.29 implies that for all $\beta \in \mathrm{O}_\pi(\mathcal{A})$ we have $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$, and thus, $\alpha \in \mathrm{PSO}_\Gamma(\mathcal{A})$. □

Lemma 8.12. *For any $\mathcal{A}$ and compositional Γ, we have $\mathrm{MPO}_\Gamma(\mathcal{A}) \subseteq \mathrm{POM}_\Gamma(\mathcal{A})$.*

Proof. We will prove that $(\mathrm{PO}_\Gamma(\mathcal{A}) - \mathrm{POM}_\Gamma(\mathcal{A})) \cap \mathrm{MPO}_\Gamma(\mathcal{A}) = \emptyset$. Since, $\mathrm{MPO}_\Gamma(\mathcal{A}) \subseteq \mathrm{PO}_\Gamma(\mathcal{A})$, this implies that $\mathrm{MPO}_\Gamma(\mathcal{A}) \subseteq \mathrm{POM}_\Gamma(\mathcal{A})$. Let $\alpha \in \mathrm{PO}_\Gamma(\mathcal{A}) - \mathrm{POM}_\Gamma(\mathcal{A})$. By Lemma 8.9, $\Gamma \cup \Delta^{\mathcal{A}}_\alpha$ is consistent; we choose some maximal model π of $\Gamma \cup \Delta^{\mathcal{A}}_\alpha$. In particular, $\mathrm{O}_\pi(\mathcal{A}) \ni \alpha$. Choose some maximal model π' of Γ extending π. Then, $\alpha \notin \mathrm{O}_{\pi'}(\mathcal{A})$, since $\alpha \notin \mathrm{POM}_\Gamma(\mathcal{A})$. Choose some $\beta \in \mathrm{O}_{\pi'}(\mathcal{A})$, and thus, $\beta \in \mathrm{O}_\pi(\mathcal{A})$, by Lemma 8.9. This implies $\alpha \equiv_\pi \beta$, and thus, by Proposition 4.29, $\Gamma \cup \Delta^{\mathcal{A}}_\alpha \models_{\mathscr{L}} \alpha \equiv \beta$. This implies that

$\mathrm{Opt}_\Gamma^{\mathcal{A}}(\beta) \supseteq \mathrm{Opt}_\Gamma^{\mathcal{A}}(\alpha)$, by Lemma 8.9. Since $\pi' \in \mathrm{Opt}_\Gamma^{\mathcal{A}}(\beta) - \mathrm{Opt}_\Gamma^{\mathcal{A}}(\alpha)$, we have that $\alpha \notin \mathrm{MPO}_\Gamma(\mathcal{A})$. □

A straight-forward argument implies that $\mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{MPO}_\Gamma(\mathcal{A})$. Lemmas 8.11 and 8.12 then imply the following.

Proposition 8.13. *For any alternatives $\mathcal{A}$ and preference statements $\Gamma \subseteq \mathcal{L}$ we have $\mathrm{PSO}_\Gamma(\mathcal{A}) \subseteq \mathrm{POM}_\Gamma(\mathcal{A}) \cap \mathrm{MPO}_\Gamma(\mathcal{A})$. If Γ is compositional then $\mathrm{PSO}_\Gamma(\mathcal{A}) = \mathrm{POM}_\Gamma(\mathcal{A}) = \mathrm{MPO}_\Gamma(\mathcal{A})$.*

We consider now the relation of class $\mathrm{EXT}_\Gamma(\mathcal{A})$ to the other classes. First let us prove, the following basic property of compositions of $\mathscr{L}$-models.

Lemma 8.14. *Let $\pi_1, \ldots, \pi_k$ be a finite sequence of models, and let $\pi = \pi_1 \circ \cdots \circ \pi_k$. Then, $\mathcal{A}_{\pi_1,\ldots,\pi_k} = \mathcal{A}_\pi = \mathrm{O}_\pi(\mathcal{A})$.*

Proof. We first show that for arbitrary $\pi, \pi' \in \mathscr{L}$, $\mathrm{O}_{\pi'}(\mathcal{A}_\pi) = \mathcal{A}_{\pi\circ\pi'}$, i.e., $\mathrm{O}_{\pi'}(\mathrm{O}_\pi(\mathcal{A})) = \mathrm{O}_{\pi\circ\pi'}(\mathcal{A})$.

Consider an element $\alpha \in \mathrm{O}_{\pi'}(\mathrm{O}_\pi(\mathcal{A}))$; we will show that $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$ for every $\beta \in \mathcal{A}$, showing that $\alpha \in \mathrm{O}_{\pi\circ\pi'}(\mathcal{A})$. We have that $\alpha \in \mathrm{O}_\pi(\mathcal{A})$, which implies $\alpha \succcurlyeq_\pi \beta$. If $\alpha \succ_\pi \beta$ then $\alpha \succ_{\pi\circ\pi'} \beta$, by Lemma 4.35. Otherwise, we have $\alpha \equiv_\pi \beta$, which implies that $\beta \in \mathrm{O}_\pi(\mathcal{A})$, and thus, $\alpha \succcurlyeq_{\pi'} \beta$. Lemma 4.35 implies that $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$.

Conversely, assume that $\alpha \in \mathrm{O}_{\pi\circ\pi'}(\mathcal{A})$. Consider any $\beta \in \mathrm{O}_\pi(\mathcal{A})$. We need to show that $\alpha \succcurlyeq_{\pi'} \beta$. We have $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$. Lemma 4.34 implies that $\alpha \succcurlyeq_\pi \beta$, which implies that $\alpha \in \mathrm{O}_\pi(\mathcal{A})$, and also $\alpha \equiv_\pi \beta$. Since $\alpha \succcurlyeq_{\pi\circ\pi'} \beta$, Lemma 4.35 implies $\alpha \succcurlyeq_{\pi'} \beta$, as required.

We now prove the result by induction. It is trivial for $k = 1$. Now, $\mathcal{A}_{\pi_1,\ldots,\pi_k} = \mathrm{O}_{\pi_k}(\mathcal{A}_{\pi_1,\ldots,\pi_{k-1}})$, which by the inductive hypothesis equals $\mathrm{O}_{\pi_k}(\mathcal{A}_{\pi_1\circ\cdots\circ\pi_{k-1}})$, which equals $\mathcal{A}_{\pi_1\circ\cdots\circ\pi_k}$, by the argument above. □

The optimality class $\mathrm{EXT}_\Gamma(\mathcal{A})$ turns out also to be equivalent to $\mathrm{PSO}_\Gamma(\mathcal{A})$ when Γ is compositional.

Proposition 8.15. *Consider any $\mathcal{A}$ and compositional $\Gamma \subseteq \mathcal{L}$. Then $\mathrm{EXT}_\Gamma(\mathcal{A}) = \mathrm{PSO}_\Gamma(\mathcal{A})$.*

Proof. Proposition 8.10 implies $\mathrm{EXT}_\Gamma(\mathcal{A}) \supseteq \mathrm{PSO}_\Gamma(\mathcal{A})$. To prove the converse, suppose that $\alpha \in \mathrm{EXT}_\Gamma(\mathcal{A})$. Then there exists a sequence $\pi_1, \ldots, \pi_k$ of models of

Γ such that $\mathcal{A}_{\pi_1,\dots,\pi_k} \ni \alpha$ and for all $\beta \in \mathcal{A}_{\pi_1,\dots,\pi_k}$, $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$. By Lemma 8.14, $\alpha \in \mathrm{O}_\pi(\mathcal{A})$, where $\pi = \pi_1 \circ \dots \circ \pi_k$, and $\Gamma \models_{\mathscr{L}} \alpha \equiv \beta$ for all $\beta \in \mathrm{O}_\pi(\mathcal{A})$. Since Γ is compositional, $\pi \models \Gamma$, and thus, $\alpha \in \mathrm{PSO}_\Gamma(\mathcal{A})$. □

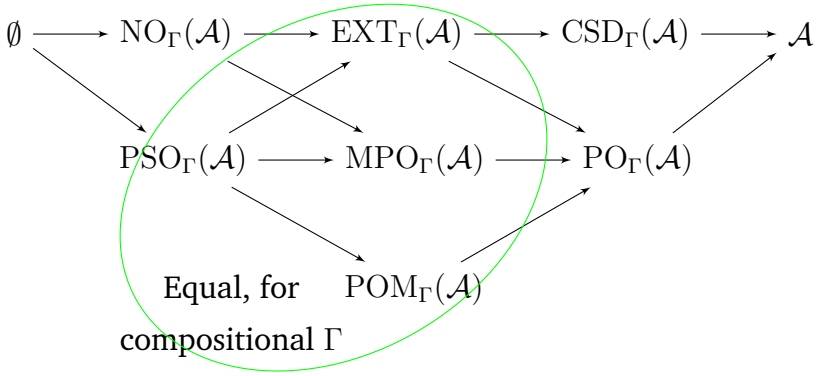
Propositions 8.10, 8.15 and 8.13 imply the following result, showing that there are substantial simplifications of the optimality classes when Γ is compositional.

Theorem 8.2: Set Relations of Different Optimality Classes

Consider any $\mathcal{A}$ and compositional $\Gamma \subseteq \mathcal{L}$. Then $\mathrm{NO}_\Gamma(\mathcal{A}) \subseteq \mathrm{PSO}_\Gamma(\mathcal{A}) = \mathrm{EXT}_\Gamma(\mathcal{A}) = \mathrm{MPO}_\Gamma(\mathcal{A}) = \mathrm{POM}_\Gamma(\mathcal{A}) \subseteq \mathrm{CSD}_\Gamma(\mathcal{A}) \cap \mathrm{PO}_\Gamma(\mathcal{A})$.

Proof. Proposition 8.13 implies that $\mathrm{PSO}_\Gamma(\mathcal{A}) = \mathrm{MPO}_\Gamma(\mathcal{A}) = \mathrm{POM}_\Gamma(\mathcal{A})$. Proposition 8.15 implies that $\mathrm{EXT}_\Gamma(\mathcal{A}) = \mathrm{PSO}_\Gamma(\mathcal{A})$. Proposition 8.10 implies that $\mathrm{NO}_\Gamma(\mathcal{A}) \subseteq \mathrm{EXT}_\Gamma(\mathcal{A}) \subseteq \mathrm{CSD}_\Gamma(\mathcal{A}) \cap \mathrm{PO}_\Gamma(\mathcal{A})$, completing the proof. □

We can summarise the relations in the following diagram, where $A \to B$ represents the relation $A \subseteq B$.



8.2.3 Computing Optimal Solutions for $\mathscr{L}$ and $\mathcal{L}'_{pqT}$

Let us now analyse the efficiency of computing PO_Γ, PSO_Γ, CSD_Γ and NO_Γ for $\Gamma \subseteq \mathcal{L}'_{pqT}$ for cvo lexicographic models $\mathscr{L}$. Note that by Theorem 8.2, since $\Gamma \subseteq \mathcal{L}'_{pqT}$ is compositional, $\mathrm{EXT}_\Gamma = \mathrm{MPO}_\Gamma = \mathrm{POM}_\Gamma = \mathrm{PSO}_\Gamma$ and thus PSO_Γ is chosen to represent all of these classes.

8.2.3.1 Membership Tests

One approach to compute $\Omega(\mathcal{A})$ for $\Omega \in \{\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma\}$ is to test membership for every alternative separately, i.e., checking if $\alpha \in \Omega(\mathcal{A})$ for all $\alpha \in \mathcal{A}$. Let $\tau(g, n)$ be an upper bound on the time to decide consistency for g statements and n variables. To test if $\alpha \in \mathrm{PO}_\Gamma(\mathcal{A})$, we test whether $\Gamma \cup \{\alpha \geq \beta \mid \beta \in \mathcal{A} - \{\alpha\}\}$ is consistent in $\tau(g + m - 1, n)$, where $|\mathcal{A}| = m$. Similarly, we test if $\alpha \in \mathrm{PSO}_\Gamma(\mathcal{A})$ in $\tau(g + m - 1, n)$, by checking if $\Gamma \cup \{\alpha > \beta \mid \beta \in \mathcal{A}, \beta \not\equiv_\Gamma \alpha\}$ is consistent. Note that by Proposition 4.32, $\beta \not\equiv_\Gamma \alpha$ can be checked by computing a maximal model π of Γ and checking $\beta \not\equiv_\pi \alpha$, which can be done in $\mathcal{O}(n^2)$ by the algorithms from Section 8.1.3. To test if $\alpha \in \mathrm{CSD}_\Gamma(\mathcal{A})$, we check for all $\beta \in \mathcal{A}$ with $\beta \not\equiv_\Gamma \alpha$ if $\Gamma \cup \{\alpha > \beta\}$ is consistent in $m\tau(g + 1, n)$. To test if $\alpha \in \mathrm{NO}_\Gamma(\mathcal{A})$, we test for all $\beta \in \mathcal{A} - \{\alpha\}$ if $\Gamma \cup \{\alpha < \beta\}$ is consistent in $m\tau(g + 1, n)$.

Let $M_\Omega(m)$ denote the worst case running time of testing if an outcome $\alpha \in \mathcal{A}$ is in $\Omega(\mathcal{A})$ for $|\mathcal{A}| = m$ and operator Ω. Then the running time to compute $\Omega(\mathcal{A})$ can be estimated by $O(mM_\Omega(m))$. Upper bounds on the theoretical running times for computing $\Omega(\mathcal{A})$ for $\Omega \in \{\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma\}$ are summarised by the following table.

	$\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma$	CSD_Γ	NO_Γ
Running time	$m\ \tau(g + m - 1, n)$	$m^2\ \tau(g + 1, n)$	$m^2\ \tau(g + 1, n)$

Consider the computation of operators $\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma$ for cvo lexicographic models $\mathscr{L}$. By the results in Section 8.1.3, we can decide $\mathscr{L}$-consistency for g statements and n variables in $\mathcal{O}(n^2 g)$. The theoretical running times for computing $\Omega(\mathcal{A})$ for $\Omega \in \{\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma\}$ are thus as summarised in the following.

	$\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma$	CSD_Γ	NO_Γ
Running time	$\mathcal{O}(mn^2(g + m))$	$\mathcal{O}(m^2 n^2 g)$	$\mathcal{O}(m^2 n^2 g)$

8.2.3.2 Incremental Approaches

In this section, we describe incremental approached to compute operators $\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma$. For this purpose, we define the notion of optimality operators as in [WRM15] and show that the considered operators are either optimality operators or satisfy similar properties.

Definition 8.4: Optimality Operators

A function $\Omega : 2^{\mathcal{A}} \longrightarrow 2^{\mathcal{A}}$ is called optimality operator over the finite set $\mathcal{A}$, if for arbitrary $A, B \subseteq \mathcal{A}$:

(I) $\Omega(A) \subseteq A$,

(II) if $A \subseteq B$ then $\Omega(B) \cap A \subseteq \Omega(A)$ and

(III) if $\Omega(B) \subseteq A \subseteq B$ then $\Omega(A) = \Omega(B)$.

We say a function $\Omega : 2^{\mathcal{A}} \longrightarrow 2^{\mathcal{A}}$ satisfies path independence if for arbitrary $A, B \subseteq \mathcal{A}$ $\Omega(A \cup B) = \Omega(\Omega(A) \cup B)$.

Note that as described in [WRM15], path independence of an operator Ω is equivalent to Ω being an optimality operator. They also make a statement for general definitions of PO and CSD. Let $\mathcal{A}$ be a set of alternatives, and $\mathcal{S}$ a set of total preorders on $\mathcal{A}$. Define $\mathrm{PO}_{\mathcal{S}}(\mathcal{A})$ to be the set of $\alpha \in \mathcal{A}$ such that there exists $\geq \in \mathcal{S}$ and for all $\beta \in \mathcal{A}$, $\alpha \geq \beta$. Define $\mathrm{CSD}_{\mathcal{S}}(\mathcal{A})$ to be the set of $\alpha \in \mathcal{A}$ such that for all $\beta \in \mathcal{A}$ with $\beta \not\equiv_{\mathcal{S}} \alpha$, there exists $\geq \in \mathcal{S}$ with $\alpha \geq \beta$.

Proposition 8.16 (*from Proposition 3 in [WRM15]*)**.** *Let $\mathcal{S}$ be a set of total preorders on some alternatives $\mathcal{A}$. The operators $\mathrm{CSD}_{\mathcal{S}}$ and $\mathrm{CSD}_{\mathcal{S}}$ are optimality operators for alternatives $\mathcal{A}$.*

Proposition 8.17. *The operator PSO_{Γ} for $\Gamma \subseteq \mathcal{L}'_{pqT}$ is an optimality operator for* cvo *lexicographic models $\mathscr{L}$.*

Proof. By Lemma 8.11, PSO_{Γ} is equal POM_{Γ}. Thus, PSO_{Γ} is equal to PO_{Γ} restricted to maximal models of Γ. By Proposition 3 in [WRM15], PO_{Γ} restricted to maximal models of Γ is an optimality operator. Thus, PSO_{Γ} is optimality operator for $\mathscr{L}$. □

The work in [WRM15] describes the algorithm "IncrementalO" to compute $\Omega(\mathcal{A})$ for optimality operators Ω and alternatives $\mathcal{A} = \{\alpha_1, \ldots, \alpha_m\}$ in an incremental way by testing if $\alpha_i \in \Omega(\Omega(\{\alpha_1, \ldots, \alpha_{i-1}\}) \cup \{\alpha_i\})$ and if so computing $\Omega(\{\alpha_1, \ldots, \alpha_i\})$ as $\Omega(\Omega(\{\alpha_1, \ldots, \alpha_{i-1}\}) \cup \{\alpha_i\})$. We formulate this algorithm in the following way.

Algorithm 8.2: Computing $\Omega(A)$ for Optimality Operator Ω

```
1. L = ∅;  D = ∅
2. FOR ( α_i ∈ A with i = 1,...,|A| ) DO
3.       IF ( α_i ∈ Ω(L ∪ {α_i}) ) THEN
4.           FOR ( β ∈ L ) DO
5.                IF ( β ∉ Ω(L ∪ {α_i}) ) THEN
6.                     D = D ∪ {β}
7.           L = (L \ D) ∪ {α_i};  D = ∅
8. RETURN L and STOP.
```

Algorithm 8.2 may be used to compute $\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma$ and CSD_Γ since all of these operators satisfy path independence and thus are optimality operators.

Under the assumption that $M_\Omega(m)$ is monotonically increasing in m, we can estimate the running time $M_\Omega(m)$ of the algorithm described in [WRM15] to compute $\Omega(m)$ by $O(\sum_{i=1,\dots,m} iM_\Omega(i))$ in the worst case and $O(mM_\Omega(1))$ in the best case.

The following proposition shows that the algorithm "IncrementalO" cannot be used to compute the class NO_Γ.

Proposition 8.18. *The class NO_Γ is not an optimality operator for* CVO *lexicographic models $\mathscr{L}$.*

Proof. We show for $A, B \subseteq \mathcal{A}$ with $\mathrm{NO}_\Gamma(B) \subseteq A \subseteq B$ that NO_Γ does not necessarily satisfy $\mathrm{NO}_\Gamma(A) = \mathrm{NO}_\Gamma(B)$ in the case of $\mathrm{NO}_\Gamma(B) = \emptyset$. Thus NO_Γ does not satisfy property (III) of optimality operators. Let B be a set of more than one alternative such that $\mathrm{NO}_\Gamma(B) = \emptyset$ and let $\alpha \in B$. For $A = \{\alpha\}$, $\mathrm{NO}_\Gamma(A) = \alpha$. Thus $\mathrm{NO}_\Gamma(A) \neq \mathrm{NO}_\Gamma(B)$ □

However, NO_Γ satisfies similar properties as optimality operators. For an operator N and $A, B \subseteq \mathcal{A}$, consider the following properties:

(I′) $N(A) \subseteq A$,

(II′) if $A \subseteq B$ then $N(B) \cap A \subseteq N(A)$,

(III′) if $N(B) \neq \emptyset$ and $N(B) \subseteq A \subseteq B$ then $N(A) = N(B)$

(IV′) if $N(A) \neq \emptyset$ then $N(A \cup B) = N(N(A) \cup B)$ and

(V′) $N(A)$ is either singleton or empty.

Even though "IncrementalO" cannot necessarily be applied to operators that are not optimality operators, we can prove that operators with properties (I′)-(V′) can be computed by the following similar algorithm.

Algorithm 8.3: Computing $N(A)$ when (I′)-(V′) holds

```
1.  L = {a_1}
2.  FOR ( α_i ∈ A with i = 2,...,n ) DO
3.        IF ( α_i ∈ N(L ∪ {α_i}) ) THEN
4.             L = {α_i}
5.        ELSE IF ( |L| > 1 ) THEN
6.                  L = L ∪ {α_i}
7.             ELSE Write L as {β}
8.                  IF ( β ∉ N({α_i, β}) ) THEN
9.                       L = L ∪ {α_i}
10. IF ( |L| > 1 ) THEN
11.      L = ∅
12. RETURN L and STOP.
```

Proposition 8.19. *Algorithm 8.3 computes $N(A)$ for any $A \subseteq \mathcal{A}$ and operator N that satisfies (I′)-(V′).*

Proof. Let L_i denote the set L after the ith iteration of the outer for-loop and let $L_1 = \{\alpha_1\}$. Let A_i denote the set of alternatives $\alpha_1, \ldots, \alpha_i$ that have been considered in the first i iterations of the outer for-loop. We prove $N(A_i) = L_i$ if $|L_i| = 1$ and $N(A_i) = \emptyset$ otherwise for $i = 2, \ldots, n$ by induction. Thus the returned set L is equal to $N(A)$.
$i = 2$:

- Suppose $N(A_2) = N(\{\alpha_1, \alpha_2\}) = \emptyset$. Then $\alpha_2 \notin N(L_1 \cup \{\alpha_2\})$ and $|L_1| = 1$ so both else-cases apply. Since also $\alpha_1 \notin N(\{\alpha_1, \alpha_2\})$, $L_2 = \{\alpha_1, \alpha_2\}$. Thus $|L_2| > 1$.
- Suppose $N(A_2) = N(\{\alpha_1, \alpha_2\}) = \{\alpha_1\}$. Then $\alpha_2 \notin N(L_1 \cup \{\alpha_2\})$ and $|L_1| = 1$ so both else-cases apply. But since $\alpha_1 \in N(\{\alpha_1, \alpha_2\})$, $L_2 = L_1$. Thus $|L_2| = 1$ and $N(A_2) = L_2$.

- Suppose $N(A_2) = N(\{\alpha_1, \alpha_2\}) = \{\alpha_2\}$. Then $\alpha_2 \in N(L_1 \cup \{\alpha_2\})$, i.e., the first if-case applies, and the algorithm sets $L_2 = \{\alpha_2\}$. Thus $|L_2| = 1$ and $N(A_2) = L_2$.

$i \to i+1$: Consider $N(A_{i+1})$.

- Suppose $N(A_{i+1}) = \emptyset$ and $N(A_i) = L_i$, i.e., $|L_i| = 1$ (by induction hypothesis). Then by (IV′), $\emptyset = N(A_{i+1}) = N(N(A_i) \cup \{\alpha_{i+1}\}) = N(L_i \cup \{\alpha_{i+1}\})$. Then $\alpha_{i+1} \notin N(L_i \cup \{\alpha_{i+1}\})$ and $|L_i| = 1$ so both else-cases apply. Furthermore, for $L_i = \{\beta\}$, $\beta \notin N(\{\beta, \alpha_{i+1}\})$. Thus, the algorithm sets $L_{i+1} = \{\beta, \alpha_{i+1}\}$ and $|L_{i+1}| > 1$.

- Suppose $N(A_{i+1}) = \emptyset$ and $N(A_i) = \emptyset$, i.e., $|L_i| > 1$ (by induction hypothesis). Then L_i is of the form $L_i = \{\alpha_j, \alpha_k, \ldots, \alpha_i\}$, where $j < k$ and j is the highest index lower i such that $L_j = \{\alpha_j\}$. Then $N(L_j \cup \{\alpha_l\}) = L_j$ for all $j \geq l \leq k$ and by induction hypothesis $N(A_l) = L_j$. We show $N(A_{i+1}) = N(L_i \cup \{\alpha_{i+1}\})$. By (IV′), $N(A_{i+1}) = N(A_k \cup \{\alpha_{k+1}, \ldots, \alpha_i\} \cup \{\alpha_{i+1}\}) = N(N(A_k) \cup \{\alpha_{j+1}, \ldots, \alpha_i\} \cup \{\alpha_{i+1}\}) = N(\{\alpha_j\} \cup \{\alpha_{k+1}, \ldots, \alpha_i\} \cup \{\alpha_{i+1}\}) = N(L_i \cup \{\alpha_{i+1}\})$. Thus, $N(L_i \cup \{\alpha_{i+1}\}) = \emptyset$ and $\alpha_{i+1} \notin N(L_i \cup \{\alpha_{i+1}\})$. Hence, the algorithm sets $L_{i+1} = L_i \cup \{\alpha_{i+1}\}$ and thus $|L_{i+1}| > 1$.

- Suppose $N(A_{i+1}) = \{\alpha_{i+1}\}$. Then because $N(A_{i+1}) \subseteq L_i \cup \{\alpha_{i+1}\} \subseteq A_{i+1}$, (III′) implies $N(L_i \cup \{\alpha_{i+1}\}) = N(A_{i+1}) = \{\alpha_{i+1}\}$. Thus, the algorithm sets L_{i+1} to $\{\alpha_{i+1}\}$. Hence, $|L_{i+1}| = 1$ and $N(A_{i+1}) = L_{i+1}$.

- Suppose $N(A_{i+1}) = \{\alpha_j\}$ for $j \leq i$. Then because $N(A_{i+1}) \subseteq A_i \subseteq A_{i+1}$, (III′) implies $N(A_i) = N(A_{i+1}) = \{\alpha_j\}$. By induction hypothesis, $L_i = \{\alpha_j\}$. Because $N(A_{i+1}) \subseteq L_i \cup \{\alpha_{i+1}\} \subseteq A_{i+1}$, (III′) implies $N(L_i \cup \{\alpha_{i+1}\}) = N(A_{i+1}) = \{\alpha_j\}$. Thus, $\alpha_{i+1} \notin N(L_i \cup \{\alpha_{i+1}\})$ and the algorithm sets $L_{i+1} = L_i = \{\alpha_j\}$. Hence, $|L_{i+1}| = 1$ and $N(A_{i+1}) = L_{i+1}$.

□

Proposition 8.20. NO_Γ *satisfies (I′)-(IV′).*

Proof. (I′): By definition of NO_Γ, $\mathrm{NO}_\Gamma(A) \subseteq A$.

(II′): Let $A, B \subseteq \mathcal{A}$ with $A \subseteq B$. Then $\mathrm{NO}_\Gamma(B) \cap A = \{\alpha \in A \mid \forall \pi \vDash \Gamma\ \forall \beta \in B : \alpha \succcurlyeq_\pi \beta\} \subseteq \{\alpha \in A \mid \forall \pi \vDash \Gamma\ \forall \beta \in A : \alpha \succcurlyeq_\pi \beta\} = \mathrm{NO}_\Gamma(A)$.

(III′): Now consider $A, B \subseteq \mathcal{A}$ with $\emptyset \neq \mathrm{NO}_\Gamma(B) \subseteq A \subseteq B$. Since $\mathrm{NO}_\Gamma(B) \subseteq A$, (II′) implies $\mathrm{NO}_\Gamma(B) \subseteq \mathrm{NO}_\Gamma(A)$. Now suppose $\mathrm{NO}_\Gamma(A) \setminus \mathrm{NO}_\Gamma(B) \neq \emptyset$ and let

$\alpha \in \mathrm{NO}_\Gamma(A) \setminus \mathrm{NO}_\Gamma(B)$. Since $\alpha \in \mathrm{NO}_\Gamma(A)$, $\forall \pi \vDash \Gamma$ and $\forall \beta \in A$, $\alpha \succcurlyeq_\pi \beta$. Since $\alpha \notin \mathrm{NO}_\Gamma(B)$, there exists $\pi' \vDash \Gamma$ and $\beta \in B \setminus A$, $\beta \succ_{\pi'} \alpha$. Let $\gamma \in \mathrm{NO}_\Gamma(B) \subseteq A$. Then $\forall \pi \vDash \Gamma$, $\gamma \succcurlyeq_\pi \beta$. In particular, $\gamma \succcurlyeq_{\pi'} \beta \succ_{\pi'} \alpha$. This is a contradiction since $\gamma \in A$ and $\alpha \in \mathrm{NO}_\Gamma(A)$. Thus $\mathrm{NO}_\Gamma(A) \setminus \mathrm{NO}_\Gamma(B) = \emptyset$, i.e., $\mathrm{NO}_\Gamma(A) \subseteq \mathrm{NO}_\Gamma(B)$.

(IV′): Let $A, B \subseteq \mathcal{A}$ with $\mathrm{NO}(A) \neq \emptyset$. Since $\mathrm{NO}_\Gamma(A) \subseteq A$, $\mathrm{NO}_\Gamma(A) \cup B \subseteq A \cup B$.

Suppose, $\mathrm{NO}_\Gamma(A \cup B) \neq \emptyset$. We show that $\mathrm{NO}_\Gamma(A \cup B) \subseteq \mathrm{NO}_\Gamma(A) \cup B$. Suppose $\mathrm{NO}_\Gamma(A \cup B) \setminus (\mathrm{NO}_\Gamma(A) \cup B) \neq \emptyset$ and $\alpha \in \mathrm{NO}_\Gamma(A \cup B) \setminus (\mathrm{NO}_\Gamma(A) \cup B)$. Then $\alpha \notin B$ and $\alpha \notin \mathrm{NO}_\Gamma(A)$ but $\alpha \in A$. Thus, there exists $\pi' \vDash \Gamma$ and $\beta \in A$ such that $\beta \succ_{\pi'} \alpha$. This is a contradiction to $\alpha \in \mathrm{NO}_\Gamma(A \cup B)$. Hence, $\mathrm{NO}_\Gamma(A \cup B) \subseteq \mathrm{NO}_\Gamma(A) \cup B \subseteq A \cup B$ and by (III′), $\mathrm{NO}_\Gamma(A \cup B) = \mathrm{NO}_\Gamma(\mathrm{NO}_\Gamma(A) \cup B)$.

Now suppose $\mathrm{NO}_\Gamma(A \cup B) = \emptyset$. We show $\mathrm{NO}_\Gamma(\mathrm{NO}_\Gamma(A) \cup B) \subseteq \mathrm{NO}_\Gamma(A \cup B)$ and thus $\mathrm{NO}_\Gamma(\mathrm{NO}_\Gamma(A) \cup B) = \emptyset$. Suppose there exists $\alpha \in \mathrm{NO}_\Gamma(\mathrm{NO}_\Gamma(A) \cup B) \setminus \mathrm{NO}_\Gamma(A \cup B)$. Then for all $\pi \vDash \Gamma$ and $\beta \in \mathrm{NO}_\Gamma(A) \cup B$, $\alpha \succ_\pi \beta$. Let $\beta \in \mathrm{NO}_\Gamma(A)$. Then for all $\pi \vDash \Gamma$ and $\gamma \in A \setminus (\mathrm{NO}_\Gamma(A) \cup B)$, $\alpha \succ_\pi \beta \succ_\pi \gamma$. Thus $\alpha \in \mathrm{NO}_\Gamma(A \cup B)$ which is a contradiction. □

Furthermore, for any $A \subseteq \mathcal{A}$ the set $\mathrm{NO}_\Gamma(A)$ is an equivalence class, i.e., if $\alpha, \beta \in \mathrm{NO}_\Gamma(A)$ then $\alpha \equiv_\Gamma \beta$. Thus, for sets $A \subseteq \mathcal{A}$ in which alternatives are pairwise non-equivalent, $\mathrm{NO}_\Gamma(A)$ is either singleton or empty. By preprocessing the set of alternatives and including only one representative of every equivalence class w.r.t. Γ-equivalence, we obtain a set of alternatives $\mathcal{A}'$ with $|\mathrm{NO}_\Gamma(\mathcal{A}')| \leq 1$ in $O(n^2 g + mn)$ time. As mentioned before, we can find Γ-equivalence classes by finding a maximal model π of Γ (in $\mathcal{O}(|\mathcal{V}|^2)$) and comparing all alternatives on the variables V_π (in $\mathcal{O}(|\mathcal{V}|\ |\mathcal{A}|)$). This enables us to use Algorithm 8.3 to compute $\mathrm{NO}_\Gamma(A)$.

The incremental computation of $\Omega(\mathcal{A})$ for $\Omega \in \{\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma, \mathrm{CSD}_\Gamma, \mathrm{NO}_\Gamma\}$ based on models $\mathscr{L}$ results in the following theoretical best and worst case running times.

	$\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma$	CSD_Γ	NO_Γ
Best case	$\mathcal{O}(mn^2g)$	$\mathcal{O}(mn^2g)$	$\mathcal{O}(mn^2g)$
Worst case	$\mathcal{O}(m^2n^2(g+m))$	$\mathcal{O}(m^3n^2g)$	$\mathcal{O}(m^2n^2g)$

We can thus expect that an incremental computation is faster than membership tests for $\mathrm{PO}_\Gamma, \mathrm{PSO}_\Gamma$ and CSD_Γ. For NO_Γ, however, the bounds on the running time are the same for the incremental approach and membership tests.

8.3 Discussion

We analysed the problems of consistency and inference based on cvo lexicographic models $\mathscr{L}$ for comparative preference languages $\mathcal{L}_{pqT}$ and $\mathcal{L}'_{pqT}$, and developed a polynomial time algorithm that runs in $O(|\Gamma||\mathcal{V}|)$, for preference statements $\Gamma \in \mathcal{L}'_{pqT}$ and variables $\mathcal{V}$. This is based on the algorithm described in Section 4.2.4.

Furthermore, we considered different notions of optimality and analysed the relations between optimality classes for the case of cvo lexicographic models $\mathscr{L}$ and compositional statements. For cvo lexicographic models $\mathscr{L}$ and statements $\mathcal{L}'_{pqT}$, we show methods of computing the defined optimality classes. This makes use of the polynomial time algorithm to solve $\mathscr{L}$-consistency. A detailed analysis of the complexities for the computational methods shows that the naive approach of testing membership for all alternatives can be outperformed by an incremental way of building up the optimal set of alternatives.

Since the considered notions of optimality can be defined for other model types, we could check if our approaches could also be applied to other model types, by checking if the corresponding operators are optimality operators or satisfy similar conditions.

Chapter 9

Conclusion

9.1 Summary

We presented approaches for preference inference based on qualitative preference models, which can be included in decision support systems to handle sparse input preference information.

Foremost, we considered expressive comparative preference statements that are relatively easy for a user to express. These include strict and non-strict versions and negations. We analysed deduction and consistency for various qualitative preference models that are based on lexicographic and Pareto orders.

We also analysed deduction and consistency under preference statements that are (strongly) compositional under some set of preference models. The concept of strong compositionality is build on properties of inference of preference statements for combinations of preference models. It is an assumption that holds true for many natural definitions of preference models and statements, as can be seen in our analysis of lexicographic, hierarchical and Pareto models. Indirectly, strong compositionality imposes some constraints on preference models, since a composition operator (with certain properties) is required to exist. However, no specific structural constraints on the preference models or preference statements are given by strong compositionality. Nonetheless, we were able to find many interesting results in this case, which ultimately leads to a general greedy algorithm to solve the Consistency Problem. It will thus be worthwhile to check strong compositionality, when exploring different models under different preference languages.

We showed that preference deduction is coNP-complete for hierarchical models, and polynomial for the case of cvo and fvo lexicographic models, where the variables are assumed to be totally ordered and value orders of variables can be fixed for all models or depending on the model. In contrast with the cvo lexicographic inference system in [Wil14], the logic developed here for lexicographic models allows strict (as well as non-strict) preference statements. The coNP-hardness result for hierarchical models is notable, since these preference logics are relatively simple ones.

Exploiting the theoretical results on properties of consistent instances for hierarchical models allows the PC-check algorithms to prune the search space much further than a MILP solver could do for the MILP formulation. The experimental results confirm that the PC-check algorithms solve the instances faster than CPLEX. Even more, the ratios between the mean solving times of the MILP and PC-check increase extremely quickly with the number of evaluations and statements.

We also examined different notions of optimality for cvo lexicographic models, and proved relationships between them. Methods to generate sets of optimal solutions for the different notions were presented together with their complexity.

For fvo singleton Pareto models and general (k-bound) Pareto models, we were able to characterise deduction and consistency through set relations of (sets of) variable sets. In the case of fvo singleton Pareto models this enables efficient polynomial algorithms. We proved that the Consistency and Deduction Problem are NP-complete for general (k-bound) Pareto models.

We conclude that efficient preference inference is possible for some types of qualitative preference models under expressive preference languages using simple (greedy) approaches, whereas other types of qualitative preference models under simple preference languages lead to NP-completeness and coNP-completeness results. The following table summarizes the complexity results by listing the membership of problems in P, NPand coNP.

Models	Language	Deduction	Consistency
$\mathcal{P}(1)$	$\mathcal{L}^{\mathcal{A}}$	P	P
$\mathcal{P}$	$\mathcal{L}^{\mathcal{A}}$	coNP	NP
$\mathcal{H}(1)$	$\mathcal{L}^{\mathcal{A}}$	P	P
$\mathcal{H}(k)$	$\mathcal{L}^{\mathcal{A}}$	coNP	NP
$\mathscr{L}$	$\mathcal{L}'_{pqT}$	P	P

9.2 Possible Future Work

Strong Compositionality There might be other common forms of preference statement that are strongly compositional, and for which the greedy algorithm from Section 4.2.4 will enable checking consistency.

Lemma 4.14 showed that the property of being strongly compositional is (roughly speaking) preserved under conjunction. Although this is far from being the case for disjunctions in general, some disjunctive statements are strongly compositional. This includes the weakly strict statements in $\mathcal{L}_{pqT}$, and restrictions on value orderings, such as being single-peaked [Con09].

It would be interesting to investigate more complex preference languages for the considered and new models, to find more examples of strongly compositional statements. Here, we could also determine under what circumstances deduction and consistency remain polynomial.

Inconsistency Bases Inconsistency bases were a helpful concept in understanding the structure of the Consistency Problem for statements $\mathcal{L}^{\mathcal{A}}$ and models $\mathcal{H}(1)$. They allowed us to find variables which cannot be included in any fvo lexicographic model that satisfies the given user preferences. Similarly, for the case of t-bound Pareto models $\mathcal{P}(t)$ with $t \geq 1$, we were able to identify variables / variable sets, which cannot be included in any model satisfying the input preferences. It would be interesting to investigate, if such structures exist for other qualitative preference models, especially, as this might also enable us to identify unsatisfiable preference statements.

Implementation and Experimental Runtime Comparison We presented many algorithmic approaches to solve consistency and deduction for different preference models and languages. Since this dissertation mostly focused on

theoretical results, we only implemented methods and compared their running times experimentally for the NP-complete case of consistency for hierarchical models. However, future work could include implementations of the remaining methods. These could also be incorporated in other systems such as multi-objective constraint optimisation problem solvers like in our paper [GRW15]. A further analysis could also involve counting the number of $\Gamma^{(\geq)}$-satisfying models and the number of models that were actually considered during the search. For the case of hierarchical models, using a relaxation of a MILP formulation as a fast check for inconsistency within the recursive approaches could be tested.

Computation of Optimality Operators We showed approaches to compute different optimality operators under CVO lexicographic models for preference statements $\mathcal{L}'_{pqT}$. These notions of optimality can also be transferred to other preference models and computed for other preference languages. We can investigate whether the optimality classes are in similar set relations for other model types and if the algorithmic approaches presented can be adapted. Furthermore, it would be interesting to compare run times of implementations experimentally for different classes and models.

Cautiousness of Preference Models We analysed preference inference for different types of preference models based on lexicographic and Pareto orders. Naturally, it would be interesting how the inferences considered compare to each other. Which models leads to "good" inference results? Our papers [GRW15] and [WG17] compare some preference models by their cautiousness, i.e., by the number of inferences made. Here it is shown that all inferences by some model types can also be made by other model types, which results in set inclusions for the sets of undominated alternatives. A broader analysis is needed that compares the here presented models, as well as other well known preference models like CP-nets, based on their cautiousness.

User Feedback While analysing preference inference for different preference statements and models is interesting from a theoretical point of view, the research on preference handling could benefit from more user studies to explore, which preference models and preference statements are realistic to present user preferences in different scenarios. Furthermore, user feedback on the quality of the inferences made for different models can be interesting.

Bibliography

[ACMM05] David J. Abraham, Katarína Cechlárová, David Manlove, and Kurt Mehlhorn. Pareto optimality in house allocation problems. In *Proc. International Symposium on Symbolic and Algebraic Computation (ISSAC) 2005*, pages 1163–1175, 2005.

[Acz48] Jean Aczél. On mean values. *Bulletin of the American Mathematical Society*, 54(4):392–400, 1948.

[AD07] Fani Athienitou and Yannis Dimopoulos. Learning CP-networks: a preliminary investigation. *Proc. 3rd Multidisciplinary Workshop on Advances in Preference Handling (PREF'07)*, 2007.

[AGJ+17] Thomas E. Allen, Judy Goldsmith, Hayden E. Justice, Nicholas Mattei, and Kayla Raines. Uniform random generation and dominance testing for CP-nets. *Journal of Artificial Intelligence Research*, 59:771–813, 2017.

[APGvG10] Pavlo Antonenko, Fred Paas, Roland Grabner, and Tamara van Gog. Using electroencephalography to measure cognitive load. *Educational Psychology Review*, 22(4):425–438, Dec 2010.

[Arr50] Kenneth J. Arrow. A difficulty in the concept of social welfare. *Journal of Political Economy*, 58(4):328–346, 1950.

[AS15] Ali E. Abbas and Zhengwei Sun. Multiattribute utility functions satisfying mutual preferential independence. *Operations Research*, 63(2):378–393, 2015.

[BA09] Linas Baltrunas and Xavier Amatriain. Towards time-dependant recommendation based on implicit feedback. In *Workshop on context-aware recommender systems (CARS'09)*, 2009.

[Bal98] Marko Balabanović. Exploring versus exploiting when learning user models for text recommendation. *User Modeling and User-Adapted Interaction*, 8(1-2):71–102, 1998.

[BBD+04a] Craig Boutilier, Ronen I. Brafman, Carmel Domshlak, Holger H. Hoos, and David Poole. CP-nets: A tool for reasoning with conditional *ceteris paribus* preference statements. *Journal of Artificial Intelligence Research*, 21:135–191, 2004.

[BBD+04b] Craig Boutilier, Ronen I. Brafman, Carmel Domshlak, Holger H. Hoos, and David Poole. Preference-based constrained optimization with CP-nets. *Computational Intelligence*, 20(2):137–157, 2004.

[BBHP99] Craig Boutilier, Ronen I. Brafman, Holger H. Hoos, and David Poole. Reasoning with conditional *ceteris paribus* preference statements. In *Proc. Uncertainty in Artificial Intelligence (UAI)*, pages 71–80, 1999.

[BCE+16] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D. Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[BCL+10] Richard Booth, Yann Chevaleyre, Jérôme Lang, Jérôme Mengin, and Chattrakul Sombattheera. Learning conditionally lexicographic preference relations. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 269–274, 2010.

[BCM16] Sylvain Bouveret, Yann Chevaleyre, and Nicolas Maudet. Fair allocation of indivisible goods. In *Handbook of Computational Social Choice*, chapter 12, pages 284–310. Cambridge University Press, 2016.

[BD02] Ronen I. Brafman and Carmel Domshlak. Introducing variable importance trade-offs into CP-nets. In *Proc. Uncertainty in Artificial Intelligence (UAI)*, pages 69–76, 2002.

[BD04] Ronen I. Brafman and Yannis Dimopoulos. Extended semantics and optimization algorithms for CP-networks. *Computational Intelligence*, 20(2):218–245, 2004.

[BD09] Ronen I. Brafman and Carmel Domshlak. Preference handling — an introductory tutorial. *AI Magazine*, 30(1):58–86, 2009.

[BDCBVL04] Edmund K. Burke, Patrick De Causmaecker, Greet V. Berghe, and Hendrik Van Landeghem. The state of the art of nurse rostering. *Journal of Scheduling*, 7(6):441–499, 2004.

[BDPP10] Denis Bouyssou, Didier Dubois, Henri Prade, and Marc Pirlot. *Decision Making Process: Concepts and Methods*. ISTE. Wiley, 2010.

[BDSS15] Tibor Besedeš, Cary Deck, Sudipta Sarangi, and Mikhael Shor. Reducing choice overload without reducing choices. *Review of Economics and Statistics*, 97(4):793–802, 2015.

[BEL09] Sylvain Bouveret, Ulle Endriss, and Jérôme Lang. Conditional importance networks: A graphical language for representing ordinal, monotonic preferences over sets of goods. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 67–72, 2009.

[BH12] Michael Bräuning and Eyke Hüllermeier. Learning conditional lexicographic preference trees. In *Preference Learning (PL-12), European Conference on Artificial Intelligence (ECAI) workshop*, pages 11–15, 2012.

[BHK98] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proc. Uncertainty in Artificial Intelligence (UAI)*, pages 43–52. Morgan Kaufmann Publishers Inc., 1998.

[BHK14] Dimo Brockhoff, Youssef Hamadi, and Souhila Kaci. Using comparative preference statements in hypervolume-based interactive multiobjective optimization. In *Learning and Intelligent Optimization. Revised Selected Papers*, pages 121–136, 2014.

[Bir17] Péter Biró. Applications of matching models under preferences. In *Trends in Computational Social Choice*, chapter 18, pages 345–373. Lulu. com, 2017.

[BJP90] James R. Bettman, Eric J. Johnson, and John W. Payne. A componential analysis of cognitive effort in choice. *Organizational Behavior and Human Decision Processes*, 45(1):111 – 139, 1990.

[BK01] Salem Benferhat and Souhila Kaci. A possibilistic logic handling of strong preferences. In *Joint 9th IFSA World Congress and 20th*

NAFIPS International Conference, 2001., volume 2, pages 962–967. IEEE, 2001.

[BKS01] Stephan Börzsönyi, Donald Kossmann, and Konrad Stocker. The skyline operator. In *Proc. International Conference on Data Engineering (ICDE)*, pages 421–430. IEEE Computer Society, 2001.

[BLW10] Meghyn Bienvenu, Jérôme Lang, and Nic Wilson. From preference logics to preference languages, and back. In *Proc. International Conference on the Principles of Knowledge Representation and Reasoning (KR)*, 2010.

[BMP09] Denis Bouyssou, Thierry Marchant, and Patrice Perny. Social choice theory and multicriteria decision aiding. In *Decision-making Process Concepts and Methods*, pages 779–810. ISTE, 2009.

[Bou02] Craig Boutilier. A POMDP formulation of preference elicitation problems. In *Proc. of the Eighteenth National Conference on Artificial Intelligence (AAAI) and Fourteenth Conference on Innovative Applications of Artificial Intelligence (IAAI)*, pages 239–246. AAAI Press / The MIT Press, 2002.

[BPRV05] Stefano Bistarelli, Maria S. Pini, Francesca Rossi, and Kristen B. Venable. Positive and negative preferences. In *Proc. International Conference on Principles and Practice of Constraint Programming (CP), workshop on preferences and soft constraints*, 2005.

[BR07] Derek Bridge and Francesco Ricci. Supporting product selection with query editing recommendations. In *Proc. ACM Conference on Recommender Systems (RecSys)*, pages 65–72, New York, NY, USA, 2007. ACM.

[BSR+05] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proc. International Conference on Machine Learning (ICML)*, pages 89–96. ACM, 2005.

[Bul14] Nils Bulling. A survey of multi-agent decision making. *KI - Künstliche Intelligenz*, 28(3):147–158, Aug 2014.

[CDS14] Daniel R. Cavagnaro and Clintin P. Davis-Stober. Transitive in our preferences, but transitive in different ways: An analysis of choice variability. *Decision*, 1(2):102–122, 2014.

[Cho54] Gustave Choquet. Theory of capacities. *Annales de l'institut Fourier*, 5:131–295, 1954.

[CL11] Richard T. Carson and Jordan J. Louviere. A common nomenclature for stated preference elicitation approaches. *Environmental and Resource Economics*, 49(4):539–559, 2011.

[Con09] Vincent Conitzer. Eliciting single-peaked preferences using comparison queries. *Journal of Artificial Intelligence Research*, 35:161–191, 2009.

[CP04] Li Chen and Pearl Pu. Survey of preference elicitation methods. *Ecole Politechnique Federale de Lausanne, Tech. Rep. IC/2004/67*, 2004.

[CP12] Li Chen and Pearl Pu. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1-2):125–150, 2012.

[CSS98] William W. Cohen, Robert E. Schapire, and Yoram Singer. Learning to order things. In *Advances in Neural Information Processing Systems*, pages 451–457, 1998.

[Dav87] Robert Davison. *Decision theory—an introduction to the mathematics of rationality, by S. French. 1986. ISBN 0-85312-682-8 (Ellis Horwood)*, volume 71. Cambridge University Press, 1987.

[DB02] Carmel Domshlak and Ronen I. Brafman. CP-nets-reasoning and consistency testing. In *Proc. International Conference on Principles of Knowledge Representation and Reasoning (KR)*, pages 121–132. Morgan Kaufmann, 2002.

[DGP09a] Jean-Philippe Dubus, Christophe Gonzales, and Patrice Perny. Fast recommendations using GAI models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1896–1901, 2009.

[DGP09b] Jean-Philippe Dubus, Christophe Gonzales, and Patrice Perny. Multiobjective optimization using GAI models. In *Proc. Inter-*

national Joint Conference on Artificial Intelligence (IJCAI), pages 1902–1907, 2009.

[DHKP11] Carmel Domshlak, Eyke Hüllermeier, Souhila Kaci, and Henri Prade. Preferences in AI: An overview. *Artificial Intelligence*, 175(7-8):1037–1052, May 2011.

[Dic18] Cambridge English Dictionary. *"preference" Meaning in the Cambridge English Dictionary*. https://dictionary.cambridge.org/dictionary/english/preference, 2018.

[DIV07] József Dombi, Csanád Imreh, and Nándor Vincze. Learning lexicographic orders. *European Journal of Operational Research*, 183(2):748–756, 2007.

[DLP94] Didier Dubois, Jérôme Lang, and Henri Prade. Possibilistic logic. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, page 439–513. Oxford Univ. Press, 1994.

[DMP+01] Didier Dubois, Jean-Luc Marichal, Henri Prade, Marc Roubens, and Régis Sabbadin. The use of the discrete Sugeno integral in decision-making: A survey. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 9(05):539–561, 2001.

[DP86] Didier Dubois and Henri Prade. Weighted minimum and maximum operations in fuzzy set theory. *Information Sciences*, 39(2):205–210, 1986.

[DRVW03] Carmel Domshlak, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Reasoning about soft constraints and conditional preferences: complexity results and approximation techniques. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 215–220, 2003.

[DSTW04] James Delgrande, Torsten Schaub, Hans Tompits, and Kewen Wang. A classification and survey of preference handling approaches in nonmonotonic reasoning. *Computational Intelligence*, 20(2):308–334, 2004.

[DVV03] Sven De Vries and Rakesh V. Vohra. Combinatorial auctions: A survey. *INFORMS Journal on Computing*, 15(3):284–309, 2003.

[FGE05] José Figueira, Salvatore Greco, and Matthias Ehrgott. *Multiple Criteria Decision Analysis—State of the Art Surveys*. International Series in Operations Research & Management Science Volume 76. Springer, 2005.

[FH10a] Johannes Fürnkranz and Eyke Hüllermeier. *Preference Learning*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010.

[FH10b] Johannes Fürnkranz and Eyke Hüllermeier. Preference learning and ranking by pairwise comparison. In *Preference Learning*, pages 65–82. Springer, 2010.

[FHWW10] Eugene C. Freuder, Robert Heffernan, Richard J. Wallace, and Nic Wilson. Lexicographically-ordered constraint satisfaction problems. *Constraints*, 15(1):1–28, 2010.

[Fis70] Peter C. Fishburn. Utility theory for decision making. Technical report, Research analysis corp McLean VA, 1970.

[Fis74] Peter C. Fishburn. Exceptional paper—lexicographic orders, utilities and decision rules: A survey. *Management Science*, 20(11):1442–1471, 1974.

[Fis75] Peter C. Fishburn. Axioms for lexicographic preferences. *The Review of Economic Studies*, 42(3):415–419, 1975.

[Fis77] Peter C. Fishburn. Condorcet social choice functions. *SIAM Journal on Applied Mathematics*, 33(3):469–489, 1977.

[Fis85] Peter C. Fishburn. Interval graphs and interval orders. *Discrete Mathematics*, 55(2):135–149, 1985.

[FM07] Peter Flach and Edson T. Matsubara. A simple lexicographic ranker and probability estimator. In *Proc. European Conference on Machine Learning (ECML)*, pages 575–582, 2007.

[Fre04] Alex A. Freitas. A critical review of multi-objective optimization in data mining: a position paper. *ACM SIGKDD Explorations Newsletter*, 6(2):77–86, 2004.

[GAG13] Joshua T. Guerin, Thomas E. Allen, and Judy Goldsmith. Learning CP-net preferences online from user queries. In *Proc. Inter-*

national Conference on Algorithmic Decision Theory (ADT), pages 208–220. Springer, 2013.

[Gib73] Allan Gibbard. Manipulation of voting schemes: A general result. *Econometrica: Journal of the Econometric Society*, 41(4):587–601, 1973.

[GL02] Michel Grabisch and Christophe Labreuche. The symmetric and asymmetric Choquet integrals on finite spaces for decision making. *Statistical Papers*, 43(1):37–52, 2002.

[GL10] Michel Grabisch and Christophe Labreuche. A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid. Université Paris1 Panthéon-Sorbonne (Post-Print and Working Papers), HAL, 2010.

[GLTW08] Judy Goldsmith, Jérôme Lang, Miroslaw Truszczynski, and Nic Wilson. The computational complexity of dominance and consistency in CP-nets. *Journal of Artificial Intelligence Research*, 33:403–432, 2008.

[GPQ06] Christophe Gonzales, Patrice Perny, and Sergio Queiroz. Preference aggregation in combinatorial domains using GAI-nets. In *DIMACS-LAMSADE Workshop on Voting Theory And Preference Modelling*, volume 6, pages 165–179, 2006.

[GPR$^+$10] Mirco Gelain, Maria Silvia Pini, Francesca Rossi, Kristen B. Venable, and Nic Wilson. Interval-valued soft constraint problems. *Annals of Mathematics and Artificial Intelligence*, 58(3-4):261–298, 2010.

[GR00] Michel Grabisch and Marc Roubens. Application of the Choquet integral in multicriteria decision making. *Fuzzy Measures and Integrals-Theory and Applications*, pages 348–374, 2000.

[Gra96] Michel Grabisch. The application of fuzzy integrals in multicriteria decision making. *European Journal of Operational Research*, 89:445–456, 1996.

[Gra03] Michel Grabisch. The symmetric Sugeno integral. *Fuzzy Sets and Systems*, 139(3):473–490, 2003.

[GRW15] Anne-Marie George, Abdul Razak, and Nic Wilson. The comparison of multi-objective preference inference based on lexicographic and weighted average models. In *Proc. International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 88–95, 2015.

[GW16] Anne-Marie George and Nic Wilson. Preference inference based on pareto models. In *Proc. International Conference on Scalable Uncertainty Management (SUM)*, pages 170–183, 2016.

[GWO16] Anne-Marie George, Nic Wilson, and Barry O'Sullivan. Towards fast algorithms for the preference consistency problem based on hierarchical models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1081–1087, 2016.

[Har06] Sandra G. Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proc. of the Human Factors and Ergonomics Society Annual Meeting (HFES)*, 50(9):904–908, 2006.

[HFCB08] Eyke Hüllermeier, Johannes Fürnkranz, Weiwei Cheng, and Klaus Brinker. Label ranking by learning pairwise preferences. *Artificial Intelligence*, 172(16-17):1897–1916, 2008.

[HMG07] Ralf Herbrich, Tom Minka, and Thore Graepel. Trueskill™: a bayesian skill rating system. In *Advances in Neural Information Processing Systems*, pages 569–576, 2007.

[HWB+11] Yaniv Hanoch, Stacey Wood, Andrew Barnes, Pi-Ju Liu, and Thomas Rice. Choosing the right medicare prescription drug plan: The effect of age, strategy selection, and choice set size. *Health Psychology*, 30(6):719, 2011.

[IJH04] Sheena S. Iyengar, Wei Jiang, and Gur Huberman. How much choice is too much? : Contributions to 401 (k) retirement plans. In *Pension Design and Structure: New Lessons from Behavioral Finance. Ed. O. S. Mitchell and S. P. Utkus.* Oxford University Press, 2004.

[Jun04] Ulrich Junker. Preference-based search and multi-criteria optimization. *Annals of Operations Research*, 130(1-4):75–115, 2004.

[Kac11] Souhila Kaci. *Working with Preferences: Less Is More*. Cognitive Technologies. Springer, 2011.

[Kac12] Souhila Kaci. Characterization of positive and negative information in comparative preference representation. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 450–455, 2012.

[KB17] Marius Kaminskas and Derek Bridge. Diversity, serendipity, novelty, and coverage: a survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 7(1):2, 2017.

[Kie02] Werner Kießling. Foundations of preferences in database systems. In *Proc. International Conference on Very Large Databases (VLDB)*, pages 311–322. Elsevier, 2002.

[KJ07] Rajeev Kohli and Kamel Jedidi. Representation and inference of lexicographic preference models and their variants. *Marketing Science*, 26(3):380–399, 2007.

[KKA10] Toshihiro Kamishima, Hideto Kazawa, and Shotaro Akaho. A survey and empirical comparison of object ranking methods. In *Preference Learning*, pages 181–201. Springer, 2010.

[Kno00] Vicki Knoblauch. Lexicographic orders and preference representation. *Journal of Mathematical Economics*, 34(2):255–267, 2000.

[KR93] Ralph L. Keeney and Howard Raiffa. *Decisions with multiple objectives: preferences and value trade-offs*. Cambridge University Press, 1993.

[Kri07] Anita Krishnakumar. Active learning literature survey. Technical report, Technical reports, University of California, Santa Cruz. 42, 2007.

[KVDT08] Souhila Kaci and Léon Van Der Torre. Preference-based argumentation: Arguments supporting multiple values. *International Journal of Approximate Reasoning*, 48(3):730–751, 2008.

[KZ10] Frédéric Koriche and Bruno Zanuttini. Learning conditional preference networks. *Artificial Intelligence*, 174(11):685–703, 2010.

[LDGS11] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, pages 73–105. Springer, 2011.

[LG03] Christophe Labreuche and Michel Grabisch. The Choquet integral for the aggregation of interval scales in multicriteria decision making. *Fuzzy Sets and Systems*, 137(1):11 – 26, 2003.

[Liu09] Tie-Yan Liu. Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.

[LL87] Martial Lacroix and Pierre Lavency. Preferences; putting more knowledge into queries. In *Proc. International Conference on Very Large Data Bases (VLDB)*, pages 217–225, 1987.

[LLCH15] Keng-Pei Lin, Chia-Yu Lai, Po-Cheng Chen, and San-Yih Hwang. Personalized hotel recommendation using text mining and mobile browsing tracking. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 191–196. IEEE, 2015.

[LM09] Jérôme Lang and Jérôme Mengin. The complexity of learning separable ceteris paribus preferences. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 848–853, 2009.

[LRHW17] Jason Lopatecki, Adam Rose, John Hughes, and Brett Wilson. Video viewer targeting based on preference similarity, April 4 2017. US Patent 9,612,995.

[LVK10] Minyi Li, Quoc Bao Vo, and Ryszard Kowalczyk. An efficient procedure for collective decision-making with CP-nets. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 375–380. IOS Press, 2010.

[Man13] David F. Manlove. *Algorithmics of matching under preferences*, volume 2. World Scientific, 2013.

[MD04] Michael McGeachie and Jon Doyle. Utility functions for ceteris paribus preferences. *Computational Intelligence*, 20(2):158–217, 2004.

[MLB15] Kinda El Maarry, Christoph Lofi, and Wolf-Tilo Balke. Crowdsourcing for query processing on web data: A case study on the skyline operator. *Journal of Computing and Information Technology (CIT)*, 23(1), 03/2015 2015.

[MP13] Sanjay Modgil and Henry Prakken. A general account of argumentation with preferences. *Artificial Intelligence*, 195:361–397, 2013.

[MRK06] Sean M. McNee, John Riedl, and Joseph A. Konstan. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *Extended Abstracts, Conference on Human Factors in Computing Systems (CHI)*, pages 1097–1101. ACM, 2006.

[MRW13] Radu Marinescu, Abdul Razak, and Nic Wilson. Multi-objective constraint optimization with tradeoffs. In *Proc. International Conference on Principles and Practice of Constraint Programming (CP)*, pages 497–512, 2013.

[MW16] Mojtaba Montazery and Nic Wilson. Learning user preferences in matching for ridesharing. In *Proc. International Conference on Agents and Artificial Intelligence (ICAART)*, pages 63–73, 2016.

[O'M13] Conor O'Mahony. *Reasoning with sorted-pareto dominance and other qualitative and partially ordered preferences in soft constraints. PhD Thesis, University College Cork*. https://cora.ucc.ie/bitstream/handle/10468/1498/ OMahonyCS_PhD2014.pdf?sequence=2&isAllowed=y, 2013.

[Osb04] Martin J. Osborne. *An Introduction to Game Theory*, volume 3. Oxford University Press, 2004.

[ÖT06] Meltem Öztürk and Alexis Tsoukias. New results on interval comparison. *Annales du LAMSADEN*, (6), 2006.

[OW13] Conor O'Mahony and Nic Wilson. *Sorted-Pareto Dominance and Qualitative Notions of Optimality*, pages 449–460. Springer Berlin Heidelberg, 2013.

[Pro16] Ariel D. Procaccia. Cake cutting algorithms. In *Handbook of Computational Social Choice*, chapter 13, pages 311–329. Cambridge University Press, 2016.

[PRVW04] Steve Prestwich, Francesca Rossi, Kristen B. Venable, and Toby Walsh. Constrained CP-nets. In *Workshop on Modelling and Solving Problems with Constraints, European Conference on Artificial Intelligence (ECAI)*, pages 101–115, 2004.

[PTA+07] Tapio Pahikkala, Evgeni Tsivtsivadze, Antti Airola, Jorma Boberg, and Tapio Salakoski. Learning to rank with pairwise regularized least-squares. In *SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, volume 80, pages 27–33, 2007.

[RDDS11] Michel Regenwetter, Jason Dana, and Clintin P. Davis-Stober. Transitivity of preferences. *Psychological Review*, 118(1):42, 2011.

[RNL+15] Jessica Rosati, Tommaso Noia, Thomas Lukasiewicz, Renato Leone, and Andrea Maurino. Preference queries with ceteris paribus semantics for linked data. In *Proc. Confederated International Conferences on On the Move to Meaningful Internet Systems (OTM)*, volume 9415, pages 423–442, Berlin, Heidelberg, 2015. Springer-Verlag.

[Sat75] Mark A. Satterthwaite. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory*, 10(2):187 – 217, 1975.

[Sch86] David Schmeidler. Integral representation without additivity. *Proc. American Mathematical Society*, 97(2):255–261, 1986.

[Sen70] Amartya Sen. *Collective Choice and Social Welfare*. Mathematical Economics Texts Series. Holden-Day, San Francisco, 1970.

[SK09] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, 2009:4:2–4:2, January 2009.

[SKP11] Kostas Stefanidis, Georgia Koutrika, and Evaggelia Pitoura. A survey on representation, composition and application of pref-

erences in database systems. *ACM Transactions on Database Systems (TODS)*, 36(3):19, 2011.

[SM06] Michael Schmitt and Laura Martignon. On the complexity of learning lexicographic strategies. *Journal of Machine Learning Research*, 7(Jan):55–83, 2006.

[SRCP06] Igor Sampaio, Geber Ramalho, Vincent Corruble, and Ricardo Prudêncio. Acquiring the preferences of new users in recommender systems-the role of item controversy. In *Proc. European Conference on Artificial Intelligence (ECAI) workshop on recommender systems*, pages 107–110, 2006.

[SS11] Arthur Schram and Joep Sonnemans. How individuals choose health insurance: An experimental analysis. *European Economic Review*, 55(6):799–819, 2011.

[Sti87] Joseph E. Stiglitz. Pareto efficient and optimal taxation and the new new welfare economics. In *Handbook of Public Economics*, volume 2, pages 991–1042. Elsevier, 1987.

[Sug74] Michio Sugeno. *Theory of fuzzy integrals and its applications*. PhD thesis, Tokyo Institute of Technology, 1974.

[Tes89] Gerald Tesauro. Connectionist learning of expert preferences by comparison training. In *Advances in Neural Information Processing Systems*, pages 99–106, 1989.

[Tho16] William Thomson. Introduction to the theory of fair allocation. In *Handbook of Computational Social Choice*, chapter 11, pages 261–283. Cambridge University Press, 2016.

[Tia09] Guoqiang Tian. Implementation of pareto efficient allocations. *Journal of Mathematical Economics*, 45(1-2):113–123, 2009.

[Tor10] Vicenç Torra. Learning aggregation operators for preference modeling. In *Preference Learning*, pages 317–333. Springer, 2010.

[TWBR11] Walid Trabelsi, Nic Wilson, Derek Bridge, and Francesco Ricci. Preference dominance reasoning for conversational recommender systems: a comparison between a comparative prefer-

ences and a sum of weights approach. *International Journal on Artificial Intelligence Tools*, 20(4):591–616, 2011.

[Ull84] Jeffrey D. Ullman. *Principles of Database Systems*. Galgotia publications, 1984.

[VB11] Paolo Viappiani and Craig Boutilier. Recommendation sets and choice queries: there is no exploration/exploitation tradeoff! In *Proc. AAAI Conference on Artificial Intelligence*, 2011.

[VdBBDB+13] Jorne Van den Bergh, Jeroen Beliën, Philippe De Bruecker, Erik Demeulemeester, and Liesje De Boeck. Personnel scheduling: A literature review. *European Journal of Operational Research*, 226(3):367–385, 2013.

[VG10] Shankar Vembu and Thomas Gärtner. Label ranking algorithms: A survey. In *Preference Learning*, pages 45–64. Springer, 2010.

[Wal07] Toby Walsh. Representing and reasoning with preferences. *AI Magazine*, 28(4):59, 2007.

[Wal14] Walter D. Wallis. *The Mathematics of Elections and Voting*. Springer Publishing Company, Incorporated, 2014.

[WB93] Molly Wilson and Alan Borning. Hierarchical constraint logic programming. *The Journal of Logic Programming*, 16(3-4):277–318, 1993.

[WDF+08] Jyrki Wallenius, James S. Dyer, Peter C. Fishburn, Ralph E. Steuer, Stanley Zionts, and Kalyanmoy Deb. Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead. *Management Science*, 54(7):1336–1349, 2008.

[WG17] Nic Wilson and Anne-Marie George. Efficient inference and computation of optimal alternatives for preference languages based on lexicographic models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1311–1317, 2017.

[WGO15] Nic Wilson, Anne-Marie George, and Barry O'Sullivan. Computation and complexity of preference inference based on hierarchical models. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 3271–3277, 2015.

[WGO17] Nic Wilson, Anne-Marie George, and Barry O'Sullivan. Preference inference based on hierarchical and simple lexicographic models. *IfCoLog Journal of Logics and their Applications*, 4:1997–2038, 2017.

[Wil04a] Nic Wilson. Consistency and constrained optimisation for conditional preferences. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 888–892, 2004.

[Wil04b] Nic Wilson. Extending CP-nets with stronger conditional preference statements. In *Proc. Nineteenth National Conference on Artificial Intelligence (AAAI), Sixteenth Conference on Innovative Applications of Artificial Intelligence (IAAI), July 25-29, 2004, San Jose, California, USA*, pages 735–741, 2004.

[Wil09] Nic Wilson. Efficient inference for expressive comparative preference languages. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 961–966, 2009.

[Wil11] Nic Wilson. Computational techniques for a simple theory of conditional preferences. *Artificial Intelligence*, 175(7-8):1053–1091, 2011.

[Wil14] Nic Wilson. Preference inference based on lexicographic models. In *Proc. European Conference on Artificial Intelligence (ECAI)*, pages 921–926, 2014.

[WM16] Nic Wilson and Mojtaba Montazery. Preference inference through rescaling preference learning. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2203–2209, 2016.

[WO11] Nic Wilson and Conor O'Mahony. The relationships between qualitative notions of optimality for decision making under logical uncertainty. In *Proc. Irish Conference on Artificial Intelligence and Cognitive Science (AICS)*, pages 66–75, 2011.

[WRM15] Nic Wilson, Abdul Razak, and Radu Marinescu. Computing possibly optimal solutions for multi-objective constraint optimisation with tradeoffs. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 815–821. AAAI Press, 2015.

[Yag88] Ronald R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems Man and Cybernetics*, 18(1):183–190, January 1988.

[YWLdJ10] Fusun Yaman, Thomas J. Walsh, Michael L. Littman, and Marie des Jardins. Learning lexicographic preference models. In *Preference Learning*, pages 251–272. 2010.

[YXYG16] Zhiwen Yu, Huang Xu, Zhe Yang, and Bin Guo. Personalized travel package with multi-point-of-interest recommendation based on crowdsourced user footprints. *IEEE Transactions on Human-Machine Systems*, 46(1):151–158, 2016.

[ZHC+04] Hua-Jun Zeng, Qi-Cai He, Zheng Chen, Wei-Ying Ma, and Jinwen Ma. Learning to cluster web search results. In *Proc. International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 210–217, 2004.