FACULTEIT ECONOMIE EN
BEDRIJFSWETENSCHAPPEN

KU LEUVEN

TESTING THEORIES OF CHOICE BEHAVIOUR

Proefschrift Voorgedragen tot
het Behalen van de Graad van
Doctor in de Toegepaste
Economische Wetenschappen

door

Bart SMEULDERS

NUMMER 479 2015

Doctoral Committee

Advisor:	
Prof. Dr. Frits C. R. Spieksma	*KU Leuven*
Co-advisors:	
Prof. Dr. Laurens Cherchye	*KU Leuven*
Prof. Dr. Bram De Rock	*Université Libre de Bruxelles*
Members:	
Prof. Dr. Yves Crama	*Université de Liège*
Prof. Dr. Federico Echenique	*California Institute of Technology*
Prof. Dr. Michel Regenwetter	*University of Illinois*
Prof. Dr. Gerhard Woeginger	*Technische Universiteit Eindhoven*

Acknowledgments

Many people have helped me over the past few years. Some were directly involved with the writing of this thesis. The influence of others was more indirect, by making my time as a Phd-student thoroughly enjoyable. I would like to thank all of these people. In the next couple of pages I will attempt to do so. I hope I will not forget anyone.

It is tradition to mention the Phd-advisor first in these acknowledgements. Even if this were not the case, there is no doubt Frits Spieksma deserves this place of honour. Frits, I have gotten to know you as a kind and intelligent mentor. Whenever I needed advice, support, or simply a second opinion on anything, you found the time to help. There is hardly a page in this text which was not improved by your suggestions. I also greatly appreciate the freedom you gave me, to pursue any research question I found interesting, and the freedom to let me set my own pace. This combination of support and freedom you offer your students is a luxury and I can not thank you enough for it.

Naturally, the advisor is followed by the co-advisors, Laurens Cherchye and Bram De Rock. To find interesting Operations Research questions for this thesis, we ventured into the land of Economics. Laurens and Bram, you were excellent guides for these expeditions, I wonder where I might have ended up without your help. I am indebted to you for introducing me to these fascinating topics and thank you for all the support you have given me.

Next, I have to thank Mike Regenwetter. I'm still a bit surprised about what our short meeting in Berlin led to. My stay in Urbana-Champaign was one of the high points of the last few years. I am very grateful for all the work you did to arrange the visit and even more so for the hospitality you showed while I was in Illinois. I also thank you for the thorough reading of my thesis, and the many comments and suggestions you made.

I would also like to thank the other members of my doctoral committee, Yves Crama, Federico Echenique and Gerhard Woeginger. I appreciate the effort you all put into reading my thesis. Your remarks have improved this text a lot.

There are two more persons who were directly involved with my thesis. First, Fabrice Talla Nobibon, in the first few years I was in Leuven, you seemed to be an extra advisor. I owe you a great deal of thanks for the time you took to discuss things with me. It was always a pleasure to work together with you and I hope we'll be able to do so again in the future. Finally, Clintin Davis-Stober, our meetings so far were short, but they have led to some of my favourite sections in this thesis.

Having thanked all of the people who helped shape this text, it is now time to thank all the people who shaped my time as a PhD student, the co-workers, friends and family. Two of these have been especially important for my academic life: Roel Leus, the advisor of my Masters thesis, and Kris Coolen, the assistant advisor. The work on my Masters thesis convinced me I wanted to pursue a PhD and this is partially due to your support. Thank you both !

Now let us move on to the other colleagues. When I arrived at the Hogenheuvelcollege, I found a group of friends, who were more than willing to welcome new students into their midst. Over the years some left, others joined. The atmosphere however, stayed the same. We shared soup, cake, lunch or simply coffee breaks, post-seminar dinners, evenings in Leuven, sports and many other things. The EURO conference in Rome ranks pretty high on my list of favourite trips. It's been a fun couple of years, and I want

to thank all of you for that. I will begin with my fellow 'nerds' of the fifth floor: Ward, Annette, Bart (II), Dries, Vikram, Sofie and Wenchao. We know there is no shame in enjoying an NP-COMPLETENESS proof. Some of the following fourth floor people do think it's a bit weird, but I like them all the same: Mieke, Yannick, Ann, Stefan, Jeroen, Pieter, Jorne, Michael, Dennis, Gert, Patricio, Morteza, Carla, Raïsa, Hamed, Valeri, Joeri and Stef. Even though I was only in Illinois for a short while, I can not forget Chris, who showed me horse racing does not have to be boring (when combined with American barbecue and betting).

Ma, pa, tenslotte wil ik jullie bedanken, omdat ik altijd op jullie kan rekenen. Zonder jullie stond ik hier nu niet. Sara-Anne, ook jou wil ik hier niet vergeten. Ik hou van je, en ben blij dit moment met je te mogen delen.

Bart Smeulders Leuven, March 2015.

Table of contents

Chapter 1

Literature Overview

1.1 Introduction

We live in a world full of choices. Before we step outside the door in the morning, we have already chosen what to eat for breakfast and which clothes to wear. For the morning commute, we decide how to travel, by what route, and whether we'll pick up coffee along the way. Dozens of small choices are made before it's even time for lunch, and then there are the less frequent, but more important decisions like buying a car, moving to a new home or setting up retirement savings. It is no wonder that decision making is a popular research topic. Within neoclassical economics, it is hypothesized that choices are made so as to maximize utility. Given this hypothesis, it follows that each choice tells us something about the decision maker, it reveals information about her underlying utility function and preferences. As we observe a decision maker over time, we can piece together more and more information. Given this information, a number of questions naturally arise:

i) Does there exist a utility function which is consistent with all the information we have ?

ii) If such a utility function exists, is it of a specific class ?

iii) If no such utility function exists, how close is the information to being consistent ?

We look at revealed preference theory to answer these questions, i.e., we are interested in characterizations of the models of behaviour that do not rely on any functional specification. This approach allows for direct tests of the decision models, without the risk that functional misspecifications lead to rejections of the model. Revealed preference characterizations are defined as conditions on the observed choices of decision makers. In this chapter, we focus on the computational aspects of revealed preference tests. In particular, we look at computational methods, i.e., algorithms that test whether data satisfies the revealed preference conditions. We also look at the tractability, i.e., the computational complexity of answering these questions, and we focus on the worst-case time-bounds of algorithms for such questions. Thus we are interested in whether a particular question is easy (solvable in polynomial time) or hard (NP-COMPLETE), and what the best-known method is for answering such a question. For two other recent overviews on revealed preference, we refer to Varian (2006) and Crawford and De Rock (2014).

Let us proceed by first motivating this computational point of view. In a very general way, it is clear that computation has become more important in all aspects of science, and economics is no exception. This is reflected in the scientific literature where computational challenges are explicitly mentioned. We illustrate this view with three quotes from recent papers.

Echenique et al. (2011)

> "Given [that calculating money pump costs can be a huge computational task], we check only for violations of GARP that involve cycles of limited length: lengths 2, 3, and 4."

Choi et al. (2014) (In the online appendix)

> "Since the algorithm is computationally very intensive, for a small number of subjects we report upper bounds on the consistent set."

Kitamura and Stoye (2014)

> "It is computationally prohibitive to test stochastic rationality on 25 periods at once. We work with all possible sets of eight consecutive periods, a problem size that can be very comfortably computed."

Another trend that supports computation in the domain of revealed preference is the ever-increasing size of datasets. It is becoming more and more common to track purchases of individual consumers/households, which can give researchers datasets far beyond the size laboratory experiments can provide. This only reinforces the need for efficient methods, such that conclusions can be drawn from such datasets.

Because of such issues, there is a quickly growing body of work on computation and economics. In this chapter, we wish to give an overview specifically of computation and revealed preference. To make this work broadly accessible, we will begin this survey by briefly introducing both revealed preference in Section 1.1.1 and some key concepts from complexity theory in Section 1.1.2. The main part of this text will then be a tour of the areas in which revealed preference methods have been applied. To begin, we will look at the classic revealed preference setting, with a single decision maker in Section 1.2. We include a subsection on goodness-of-fit and power measures, which respectively quantify the severity of violations and give a measure of how stringent the tests are. Next, we explore collective settings, where the observed choices are the result of a joint decision (Section 1.3). Furthermore, we look at stochastic preferences. In this setting, the decision maker still chooses so as to maximize utility, but preferences are not fixed. Instead, the decision maker has a number of different utility

functions, and which of these utility functions she maximizes at any given time is probabilistic. We handle this setting in section 1.4.

To close this subsection, we wish to make a note on the use of the words *dataset* and *data* throughout this thesis. Within social sciences, these terms usually refer to empirical observations of behaviour. While the main goal of this thesis is to study methods for analysing observed behaviour, we will use a broader definition of data. Since from an algorithmic point of view, it does not matter whether the input reflects actual observed behaviour or is constructed in some other way, we refer to all input of algorithms as 'data', no matter the origin.

1.1.1 Revealed Preference

Let us now introduce the basic ideas of revealed preference, by looking at revealed preference in the simple setting of a person making purchases. Specifically, suppose we are in a world with m different goods, whose prices are denoted by the vector $p \in \mathbb{R}^m_{++}$. Given these prices and a budget, the decision maker buys a bundle of goods, given by the vector $q \in \mathbb{R}^m_+$. (It is generally assumed the budget available to the decision maker is equal to $p \times q$. In what follows we will shorten this product to pq.) Observing these prices and quantities at n different points in time gives a dataset $S = \{(p_i, q_i) | i \in N\}$ with $N = \{1, \ldots, n\}$. We use the word *observation* to denote the information $(p_i, q_i), i \in N$. If we now suppose the decision maker has a well-behaved (concave, continuous and strictly monotone) utility function $u(q) : \mathbb{R}^m_+ \to \mathbb{R}_+$, which she attempts to maximize, then knowledge of an observation (p_i, q_i) gives a lot of information about u. For example, we know that the bundle q_i must give her at least as much utility as any other bundle with an equal or lower cost given prices p_i. If not, she is not maximizing her utility. Then, in terms of revealed preference, we will say she prefers the chosen alternative over the other alternatives, formalized by the *direct revealed preference relation*.

Definition 1.1.1. ***Direct Revealed Preference Relation***

For any pair of observations $i, j \in N$, if $p_i q_i \geq p_i q_j$, we say that q_i is revealed preferred over q_j, or $q_i \; R_0 \; q_j$.

It is important to note that if a person chooses a bundle q_i while an alternative q_j was available for the same price, she does not necessarily think q_i gives her more utility than q_j. It may also be the case that she is indifferent between the two. However, if the bundle q_i is more expensive than the bundle q_j (at prices p_i), then choosing q_i gives more information. Since saving money also has positive utility, the decision maker thinks q_i will give her more utility than q_j, in other words, that q_i is strictly better than q_j. This is formalized in the *strict direct revealed preference relation.*

Definition 1.1.2. ***Direct Strict Revealed Preference Relation***
For any pair of observations $i, j \in N$, if $p_i q_i > p_i q_j$, we say that q_i is strictly revealed preferred over q_j, or $q_i \; P_0 \; q_j$.

Because of the nature of utility and preferences, it is also important to take transitivity into account. Indeed, if a person at one point in time makes a choice revealing she values q_i more than q_j, and at another point in time she reveals to prefer q_j over q_k, then if she has a consistent utility function, she should also value q_i over q_k. As such, we define the *revealed preference relation.*

Definition 1.1.3. ***Revealed Preference Relation***
For any set of bundles $q_i, q_j, \ldots, q_k$, $i, j, \ldots, k \in N$, if $q_i \; R_0 \; q_j \; R_0 \; \ldots \; R_0 \; q_k$, we say that q_i is revealed preferred over q_k and we write $q_i \; R \; q_k$.

We have now shown what kind of information can be learned from observing choices by a decision maker. Now suppose we have a dataset of purchasing decisions $S = \{(p_i, q_i) | i = 1, \ldots, n\}$, from which we extract all of these revealed preference relations. A question then arises: is there some utility function $u(q)$, such that the bundles bought maximize this function for all observations ? This is the *rationalizability* question.

Definition 1.1.4. ***Rationalizability***
A dataset S is rationalizable by a utility function, if and only if there exists

a utility function $u(q)$, such that the observations present in the dataset are consistent with maximizing $u(q)$ under budget constraints.

A main goal of a revealed preference approach is to find conditions on the dataset such that it is rationalizable. For example, a necessary and sufficient condition for rationalizability by a well-behaved utility function is given by the *general axiom of revealed preference* (GARP).

Definition 1.1.5. ***General Axiom of Revealed Preference (GARP)*** *A dataset S satisfies GARP if and only if for each pair of distinct bundles, q_i, q_j, $i, j \in N$, if $q_i \; R \; q_j$, then it is not the case that $q_j \; P_0 \; q_i$.*

To conclude this section, we note that this rationalizability question is not limited to general utility functions. Datasets can also be rationalized by utility functions of a specific form, such as separable utility functions. Likewise, they can be rationalized by different forms of collective decision making, stochastic utility functions, or even heuristic choices.

1.1.2 Computational Complexity

We will close this introduction with a short and informal primer on computational complexity. We refer to Garey and Johnson (1979) or Cormen et al. (2001) for a more thorough introduction. We will look at two important topics in complexity. First, we consider the analysis of algorithms, in particular their *worst-case complexity*. Second, we will look at the theoretical complexity of computational problems.

It is clear that an algorithm for a computational problem performs a number of elementary operations such as addition, multiplication, comparison, etc. Also, it is clear that the larger an instance of a problem, the more of these operations will be needed to solve the problem. Thus, we see the number of operations needed by the algorithm as a function of the size of the input. In our case, the input consists of a dataset S, whose size is

measured in the number of observations n. (There is also the number of goods m, but this is rarely an important factor in computation time, and as such is usually ignored.) Next, we let the worst-case time-complexity of an algorithm be the maximum number of operations the algorithm needs to solve the problem for any dataset of size n. We will use Big O notation to describe this time-complexity of an algorithm (Cormen et al., 2001). Big O notation characterizes functions by their growth rate, leaving out constant factors and smaller terms. For example, if we are given a dataset S consisting of n observations, we might have an algorithm which needs at most $3n^3 + n^2$ operations to test whether GARP holds. Since both the n^2 term and the factor 3 have a negligible impact on the increase in computation time as n increases, we will say this algorithm has a worst-case time-complexity of $O(n^3)$.

The previous paragraph describes how to express the complexity of an algorithm. One can also speak of the complexity of a (decision) problem. A problem is a decision problem if an instance of this problem has a YES or NO answer. The complexity of a decision problem depends, to some extent, on the time-complexity functions of algorithms solving the problem. For instances, if a decision problem can be solved by an algorithm whose time-complexity function is polynomial, then the problem is said to be in the class P. Informally, we will call these *efficiently solvable* or *easy problems*; the computation time needed for such a problem rises relatively slowly with the size of the instance, and as such even large instances can usually be handled. P is a subclass of the class NP, which can be defined informally as follows: if an instance of a problem in the class NP has a YES-answer, then a polynomial size proof of this exists, which can be efficiently checked. For example, consider the following problem: given a dataset S, is GARP violated? This question is in NP, because if there is a violation, the sequence of at most n revealed preference relations which constitute the violation can be given as a proof. While testing GARP is easy (in P), the class NP also contains seemingly more difficult problems,

for which no polynomial time algorithms are known. However, as of today a proof that no polynomial time algorithms can exist for these problems is lacking. This has led to the notion of NP-Completeness: a problem is NP-Complete if it is in NP and at least as difficult as any other problem in NP. This is usually proven by reducing another known NP-Complete problem to it, thereby showing that if a polynomial time algorithm for the problem exists, this can be used to solve the other NP-Complete problem, (and thus all NP-Complete problems) in polynomial time. Since this is unlikely, proving a problem to be NP-Complete is a strong indication that the problem is in fact difficult. We will also refer to NP-Hardness in the remainder of the text, problems that are NP-Hard are not necessarily in NP, but are at least as difficult as the NP-Complete problems.

To conclude this section, we would like to note that for NP-Complete or NP-Hard problems, it can be the case that some special cases are easy, because of structure in the dataset. This may be relevant for much of the revealed preference literature, as almost all of the hardness results we will see are for the general case of these problems. In a recent paper, Deb and Pai (2013) show that if the number of goods in a dataset is small enough, some structure will appear. So far, we are unaware of any papers describing what kind of structure instances with low numbers of goods have, or how to exploit that structure.

1.2 Single Decision Maker

In the previous section, we have already given a quick introduction into revealed preference. In this section, we will begin by stepping back and exploring some of the history of revealed preference theory. We will start with the seminal paper by Samuelson (1938). Here, a decision maker is faced with purchasing decisions (indexed by i). Given prices, represented by the vectors $p_i \in \mathbb{R}^m_{++}$, she will buy a bundle of goods $q_i \in \mathbb{R}^m_+$, providing a dataset of n observations, $S = \{(p_i, q_i) | i = 1, \ldots, n\}$. As we have

seen in the previous section, all of these decisions will provide us with revealed preference relations. Using the direct revealed preference relation, Samuelson formulated the *Weak Axiom of Revealed Preference*.

Definition 1.2.1. ***Weak Axiom of Revealed Preference*** *(*WARP*)*
A dataset S satisfies WARP *if and only if for each pair of distinct bundles, q_i, q_j, $i, j \in N$, if $q_i R_0 q_j$, then it is not the case that $q_j R_0 q_i$.*

WARP is the first rationalizability condition described in the literature and requires the revealed preference relation to be asymmetric. As the axiom does not require transitivity, satisfying it is only a necessary condition for rationalizability by a single-valued utility function. A single-valued utility functions is such that, for every combination of prices and expenditures, there is only a single bundle of goods which maximizes utility. In the special case in which the dataset contains only two goods, WARP is a necessary and sufficient condition for rationalizability by a single-valued utility function (Samuelson, 1948; Little, 1949). The work of Samuelson was followed by Houthakker (1950), who extended WARP to incorporate transitivity. Using indirect revealed preference relations, he formulated the *Strong Axiom of Revealed Preference*.

Definition 1.2.2. ***Strong Axiom of Revealed Preference*** *(*SARP*)*
A dataset S satisfies SARP *if and only if for each pair of distinct bundles, q_i, q_j, $i, j \in N$, if $q_i \; R \; q_j$, then it is not the case that $q_j R_0 q_i$.*

SARP is a necessary and sufficient condition for rationalizability by a single-valued utility function for any number of goods. Rose (1958) formally proved the equivalence of WARP and SARP for two goods. Conditions for a general utility function were provided by Afriat (1967b), in the form of what is now known as the *Afriat Inequalities*, a linear programming formulation, and the *Cyclical Consistency* condition. Diewert (1973) clarified these results further and provided another linear programming formulation. Finding both the linear programs and cyclical consistency hard to test directly, Varian (1982) sought and found a condition similar to SARP

which does allow for multiple utility-maximizing bundles, GARP. As shown in the previous section, for GARP, strict preference relations are identified, and indifference is allowed if the revealed preference relations are not strict.

Having defined these core axioms of revealed preference, let us look at the computational difficulty of testing whether they are satisfied. First, WARP allows for a straightforward $O(n^2)$ test, since testing each pair of observations for a violation is sufficient. Next we turn to SARP, for which a first attempt at providing a test was made by Koo (1963). Using a matrix representation of the direct revealed preference relations, he describes a sufficient condition for consistency with SARP. Dobell (1965) was the first to describe conditions which are both necessary and sufficient. Dobell's test is also based on the matrix representation of direct revealed preference relations. He proposed checking whether every square submatrix of the direct revealed preference matrix contains at least one row and one column consisting completely of 0 elements. Due to the fact that there exist an exponential number of such submatrices, this test runs in exponential time. Koo (1971) later published another paper which is the basis for the most efficient test of SARP consistency. In it, he describes a digraph representation of revealed preference relations. In this graph, there is a node for each bundle q_i in the dataset. An arc from node i to j exists if and only if $p_i q_i \geq p_i q_j$ and $q_i \neq q_j$. Thus, to each dataset S we can associate its revealed preference graph G_S. Any cycle within the graph G_S points to a violation of SARP in the dataset S. Since testing whether a graph is acyclic can be done in $O(n^2)$, this is the most efficient known algorithm for testing consistency with SARP.

Rationalizability tests for general utility functions, or equivalently for consistency of datasets with GARP, have also gone through a number of stages, though in this case efficient and correct tests were known from the start. Afriat (1967b) provides a linear program, the Afriat Inequalities, which is polynomially solvable, although it is interesting to note that no polyno-

mial time algorithms for solving linear problems were known at the time Afriat published his work. Diewert (1973) then published a further linear program. Varian's formulation of GARP (Varian, 1982) provides another algorithm for testing rationalizability. This formulation shows rationalizability can be tested by computing the transitive closure of the matrix of direct revealed preference relations. This transitive closure gives all revealed preference relations, direct and indirect. Given the transitive closure, GARP can be tested by checking, for each pair of bundles q_i, q_j, $i, j \in N$, whether both $q_i \ R \ q_j$ and $q_j \ P_0 \ q_i$. The bottleneck in this procedure is the computation of the transitive closure. For its ease of implementation, Varian suggests using Warshall's algorithm (Warshall, 1962) to do so, which has a worst-case bound of $O(n^3)$. Varian also noted faster algorithms did exist based on matrix multiplication, which at the time reached $O(n^{2.74})$ (Munro, 1971). By now, these algorithms have improved, the best known algorithms for general matrices reaching $O(n^{2.373})$ (Coppersmith and Winograd, 1990; Williams, 2012; Gall, 2014). In Chapter 2, we describe an algorithm with a lower worst-case bound of $O(n^2)$, based on the graph representation of Koo and the computation of strongly connected components. We also prove a lower bound on testing GARP, showing no algorithm can exist that is faster than $O(n \log n)$ time.

To finish our overview on rationalizability by a general utility function, we note recent work on indivisible goods and non-linear budget sets. Fujishige and Yang (2012) and Polisson and Quah (2013) extend the revealed preference results to the case with indivisible goods. They find that GARP is a necessary and sufficient test for rationalizability, given a suitable adaptation of the revealed preference relations for their setting. Forges and Minelli (2009) give a revealed preference characterization for non-linear budgets, for which GARP is a sufficient and necessary condition for rationalizability by a locally non-satiated utility function. Cherchye et al. (2014a) give a characterization for an increasing, concave and continuous utility function for the setting with non-linear budgets in the form of a set of linear

equalities. Cosaert and Demuynck (2013) look at choice sets which are non-linear and have a finite number of choice alternatives. They provide revealed preference characterizations for weakly monotone, strongly monotone, weakly monotone and concave, and strongly monotone and concave utility functions, all of which are easy to test, either by some variant of GARP or a system of linear inequalities.

Besides the basic tests discussed in the previous paragraphs, conditions and tests have been derived for testing rationalizability by various specific forms of utility functions. First, we look at *weakly separable utility functions*. A utility function $u(q)$ is separable, if the vector of goods q can be split into at least two subvectors q^1 and q^2 and a subutility function $f(q^2)$ exists, such that $u(q) = \bar{u}(q^1, f(q^2))$. Following his paper on general utility functions, Afriat also wrote an unpublished work on separable utility functions (Afriat, 1967a). Varian (1983) built further on this, giving a non-linear system of inequalities, for which the existence of a solution is a necessary and sufficient condition for rationalizability by a concave weakly separable utility function. Diewert and Parkan (1985) extended this result to multiple separable subsets. Unfortunately, the non-linear systems resulting from Varian's and Diewert and Parkan's work are difficult to test. An implementation by Varian attempts to overcome the computational difficulties by finding a solution to the linear part of the system of inequalities and then fixing variables based on this solution, which linearises the remainder of the inequalities. This implementation can be too restrictive, as the variables are usually fixed with values making the system infeasible, even if a solution exists, as shown by Barnett and Choi (1989). Fleissig and Whitney (2003) take a similar approach, but improve on it by fixing variables with values that are more likely to allow solutions to the rest of the system of equalities. Exact tests of (adaptations of) Varian's inequalities are described in Swofford and Whitney (1994) and Fleissig and Whitney (2008). Both use non-linear programming packages to find solutions and are limited in the size of datasets they can handle.

The difficulty of the problem is established by Cherchye et al. (2014b), who prove NP-HARDNESS of the rationalizability question. Additionally, they provide an integer programming formulation which is equivalent to Varian's non-linear set of inequalities. Closely related, Echenique (2014) proves rationalizability is also NP-HARD for weakly separable utility functions when dropping the concavity assumption and even if the dataset is limited to 9 goods. Quah (2012) provides a testing algorithm for weakly separable utility functions without concavity assumption. Other works on weakly separable utility functions include Swofford and Whitney (1994), who modify the system of inequalities of Varian to account for consumers needing time to adjust their spending.

For strong or additive separability ($u(q^1, q^2) = f^1(q^1) + f^2(q^2)$), Varian (1983) gives a linear programming formulation, allowing for tests in polynomial time. Furthermore, a number of results have been published on *homothetic utility functions*. These functions are of the form $u(q) = f(l(q))$, with l a homogeneous function and f monotonically increasing. In effect, if for two bundles q_i, q_j, $u(q_i) \geq u(q_j)$, then for any constant $\alpha > 0$, $u(\alpha q_i) \geq u(\alpha q_j)$. Afriat (1972) was the first to provide a system of linear inequalities for which the existence of a solution is a necessary and sufficient condition for rationalizability by a homothetic utility function. In this formulation, the number of inequalities is exponential. Varian (1983) proposes an alternative test, which is equivalent to finding a negative length cycle in a graph (this is not the graph G_S described earlier). In the same paper he also provides a test for homothetic, separable utility functions, which is again a difficult-to-solve system of non-linear inequalities. Finally, utility maximization in case of rationing (i.e., there are additional constraints on the bundles which can be bought, on top of the budget constraint) is also handled by Varian. He provides a linear system of inequalities, for which the existence of a feasible solutions is a necessary and sufficient condition for rationalizability.

As we have seen, various forms of utility functions usually have an associated system of inequalities, for which the existence of a solution is a necessary and sufficient condition for rationalizability by such a utility function. The difficulty of these rationalizability tests generally depends on whether these systems have linear or non-linear inequalities. General, single-valued and strongly separable utility functions are easy to rationalize, as their associated systems of inequalities are linear. The same holds true for utility maximization by a general utility function under rationing constraints. For general and single-valued utility functions, more straightforward tests were developed. A polynomial test also exists for rationalizability by a homothetic utility function. For utility functions for which the associated system of inequalities are non-linear, weakly separable and homothetic separable functions, no efficient tests are known. For weakly separable utility, formal NP-HARDNESS results exist. For homothetic separable functions, this remains an open question.

1.2.1 Goodness-of-Fit and Power Measures

An often cited limitation of rationalizability tests is that they are binary tests: either the dataset is rationalizable or it is not. Thus, if violations of rationalizability conditions are found, there is no indication of how severe they are. Likewise, if the rationalizability conditions are satisfied, this could be because the choices faced by the decision maker make it unlikely that violations would occur. To get this information, so-called goodness-of-fit measures and power measures have been proposed in the literature. Goodness-of-fit measures quantify the severity of violations, while power measures give a measure of how far removed the choices are from violating rationalizability conditions.

Goodness-of-Fit Measures

A first class of goodness-of-fit measures are based on the systems of inequalities which are used to describe rationalizability conditions for many different forms of utility functions, as described earlier in this section. Extra slack variables are added to these systems, which relaxes the constraints on the data. An optimization problem can be defined, for which the objective function is the minimization of the value of the slack variables, under the constraint that the system of equalities is satisfied. The goodness-of-fit measure is then equal to the value of the optimal solution to this optimization problem. Such an approach was first described by Diewert (1973) and has since been used in a number of different papers for various forms of utility functions (See Diewert and Parkan (1985); Fleissig and Whitney (2005, 2008) for weak separability, Fleissig and Whitney (2007) for additive separability). Minimizing these slack variables is easy if the system of inequalities is linear, which is the case for general utility functions and additive separable utility functions. In the case of non-linear systems of inequalities, minimizing the slack variables is at least as hard as finding a solution to the system without slack variables. Since this is already NP-HARD for weakly separable utility functions, the hardness result remains valid for the problem of minimizing the slack variables.

A second class of goodness-of-fit measures is due to work by Afriat (1973), and is based on relaxations of the revealed preference relations. In this relaxation, revealed preference relations are used if the difference in price between the chosen bundle and an another affordable bundle is big enough. This is done by adding *efficiency indices* $0 \leq e_i \leq 1$ for each observation $i \in N$, and defining the revealed preference relations as follows.

$$\text{For all } i, j \in N, \text{ if } e_i p_i q_i \geq p_i q_j, \text{ then } q_i \; R_0(e_i) q_j. \tag{1.1}$$

Obviously, if $e_i = 1$, conditions (1.1) are the same revealed preference relations as those described in Definition 1.1.1; if $e_i < 1$, this can be interpreted as having a revealed preference relation between two bundles if the price difference between bundles exceeds a certain fraction of the budget. As a

result, there will be fewer revealed preference relations, and axioms such as WARP, SARP and GARP will be easier to satisfy. A goodness-of-fit measure is then to maximize the sum of e_i values, under the constraint that a given axiom of revealed preference is satisfied. Goodness-of-fit measures based on this idea have been described by Afriat (1973), Varian (1990) and Houtman and Maks (1985). Of these three, Afriat's index is the simplest, as the value e_i is constrained to be equal for every observation ($e_1 = e_2 = \ldots = e_n$). This makes it easy to compute, even though for a long time the only published algorithm was an approximation algorithm due to Varian (1990). Varian's index allows e_i values to differ between observations. This makes computation less straightforward and the computation was thus perceived to be hard. This led to work on heuristic algorithms for computing Varian's index by by Varian himself (Varian, 1990), Tsur (1989) and more recently by Alcantud et al. (2010). Finally, Houtman and Maks (1985) proposed to constrain the e_i values to either 0 or 1. In effect, this means removing the minimum number of observations such that the remaining dataset is rationalizable. Houtman and Maks established a link between the known NP-HARD problem of feedback vertex set and their index for the strong axiom of revealed preference, showing its difficulty. Dean and Martin (2010) propose a weighted variant of Houtman and Maks' index. To be able to compute these measures for larger datasets, Dean and Martin make use of an equivalence to set covering problems. While this problem is also NP-HARD, it is well studied and good algorithms exist. We establish the complexity of computing all three of these indices in Chapter 3, providing polynomial time algorithms to calculate Afriat's index for various axioms of revealed preference. Varian's and Houtman and Maks' index are shown to be NP-HARD and even stronger, it is shown that no constant-factor approximation algorithms running in polynomial time exist unless P = NP.

Furthermore, we distinguish a third approach introduced by Varian (1985). If a dataset fails to satisfy the rationalizability conditions, their goal is to find a dataset which does satisfy the conditions and is only minimally dif-

ferent from the observed dataset. The problem of finding these minimally different satisfying datasets is a non-linear optimization problem, which, in general, are hard problems to solve. To avoid having to solve large scale non-linear problems, De Peretti (2005) approaches this problem with an iterative procedure. Working on GARP, his algorithm tackles violations one at a time, also perturbing only one observation at a time. If a preference cycle exists between two bundles of goods q_i and q_j, $i, j \in N$, he computes the minimal perturbation necessary to remove the violation both for the case in which $q_i \; R_0 \; q_j$ (in which case q_i is perturbed) and for the case in which $q_j \; R_0 \; q_i$ (in which case q_j is perturbed). The smallest perturbation of the two is then used to update the dataset, and the updated dataset is checked for new GARP violations. While this does not guarantee an optimal solution, it does allow for handling larger datasets, especially if the number of violations is small.

Showing the continuing interest in goodness-of-fit measures, a number of recent papers introduce new goodness-of-fit measures. Echenique et al. (2011) define the mean and median money pump indices. In this paper, the severity of violations of rationality is measured by the amount of money which an arbitrageur could extract from the decision maker by exploiting her irrational choices. This is reflected by a money pump index for every violation of rationality. Echenique et al. propose to calculate the money pump index of the mean and median violation as a measure of the irrationality of the decision maker. In Chapter 4, we show that computing these measures is NP-HARD, but that computing the money pump index for the most and least severe violation can be done in polynomial time. Furthermore, Apesteguia and Ballester (2014) introduce the minimal swaps index. Informally, the swaps index of a given preference ordering over the alternatives is calculated by counting how many better alternatives (according to the preference order) were not chosen over all choice situations. The minimal swaps index is then the the swaps index of the preference order for which this index is minimal. Apesteguia and Ballester show that computing the minimal swaps index is equivalent to the NP-HARD linear

ordering problem.

Power Measures

Power measures were first introduced by Bronars (1987). Informally, the goal of power measures is to quantify how unlikely it is that choices generated by an alternative model of behaviour satisfy rationalizability conditions for utility maximization. The more likely this is, the lower the power of the test. Indeed, suppose we have a large number of datasets for which we know that choices were not made to maximize a utility function. If these datasets can often be rationalized by a utility function, then it is obvious that the test is not good at discriminating between utility maximizing and other behaviour. Bronars proposes to use random choices as an alternative model. The likelihood of this alternative model satisfying the conditions is determined by Monte Carlo simulation. Andreoni and Miller (2002) use a similar approach, they generate synthetic datasets by bootstrapping from observed choices, and use these to establish the power of their test.

Bronars's Monte Carlo approach has also been applied to goodness-of-fit measures. The value of a Goodness-of-fit measures is hard to interpret without context. There is no natural level which, if crossed, indicates a large deviation from rational behaviour. Furthermore, the values of goodness-of-fit indices which point to large deviations may vary from dataset to dataset, as the choices faced by a decision maker may or may not allow large violations of rationalizability. One way to establish what values are significant, is to use a Monte Carlo approach and calculate the goodness-of-fit measures for these generated datasets. This gives a distribution of the values of goodness-of-fit measures for datasets of random choices. It can then be checked whether the goodness-of-fit measures found for the actual decision makers are significantly different. For example,

Choi et al. (2007) and Heufer (2012) use this approach. As this approach requires the goodness-of-fit measures to be calculated a large number of times, there is a strong incentive to use efficient algorithms and measures which are easy to calculate.

Beatty and Crawford (2011) propose measuring the power of a test by calculating the proportion of possible choices which would satisfy the conditions. Finally, we refer to Andreoni et al. (2013), who give an overview of power measures and indices and introduce a number themselves. The measures they introduce are adaptations of goodness-of-fit measures. For example, they introduce a Jittering index, which is the minimum perturbation of the data so the rationalizability conditions are no longer satisfied, in line with of the work of Varian (1985). They also introduce an Afriat Power Index, which is the reverse of the Afriat's goodness-of-fit measure, i.e., instead of finding the minimum e value such that the dataset no longer satisfies the considered axiom of revealed preference, it is the maximum e value such that the dataset does not satisfy the conditions.

1.3 Collective Choices

In the preceding section, data were analysed as if a single person bought or chose goods, to maximize her own utility function. However, in many cases purchasing decisions are observed at the household level. Analysing these data calls for collective models, which account for individually rational household members, and some decision process for splitting up the budget. The initial revealed preference contributions on this subject were published by Chiappori (1988), for the labour supply setting. Leisure time is observed for each member in the household, as well as the aggregated consumption. The behaviour of this household is then rationalizable if the consumption can be split up such that the resulting individual datasets of leisure and consumption are rationalizable for all household members.

Chiappori provides conditions for rationalizability, both for the cases with and without externalities of private consumption. For both cases, these take the form of non-linear systems of inequalities, making testing hard. Snyder (2000) provides a reformulation of Chiappori's conditions and uses it in empirical tests. Testing her reformulation in a straightforward method requires a linear integer program. Whether a polynomial test for the labour supply setting exists remains an open question.

The work by Chiappori was generalized by Cherchye et al. (2007). Leaving the labour supply setting, they provide conditions for an arbitrary number of goods and without any prior allocation of goods, as was the case with leisure time in Chiappori's work. Cherchye et al. derive separate necessary and sufficient conditions for collective rationalizability by concave utility functions. In a later paper, Cherchye et al. (2010) show that the necessary condition given in their earlier work is both necessary and sufficient, when dropping the assumption of concave utility functions. However, testing this condition is NP-HARD, as shown by Talla Nobibon and Spieksma (2010). Due to the hardness of rationalizability in collective settings, a number of papers have appeared on how to test this problem. An integer programming formulation is given by Cherchye et al. (2008) and an enumerative approach is provided by Cherchye et al. (2009). Talla Nobibon et al. (2011) take a different approach, giving a heuristic algorithm. The goal of this algorithm is to quickly test whether the rationalizability conditions are satisfied. If this heuristic can not prove the conditions are satisfied, an exact test is used. Using this heuristic pre-test, many computationally demanding exact tests can be avoided. Deb (2010) strengthens the hardness results by proving that a special case of this problem, the situation dependent dictatorship setting, is also NP-HARD. In this setting, the household decision process is such that each purchasing decision is made by a single household member, the dictator. At different points in time, different household members can be the dictator, the goal is thus to partition the observations into datasets, so that each dataset is consistent with (unitary)

GARP. Crawford and Pendakur (2013) also consider this problem in the context of preference heterogeneity, they provide algorithms for computing upper and lower bounds on the number of 'dictators'. In Chapter 6, we give further hardness results for a collective version of WARP: we find that dropping transitivity makes the test easy for households of 2 members, but the problem remains open for 3 or more.

Another branch of the literature returns to Chiappori's setting, in the sense that observers know whether specific goods are either publicly or privately consumed. Given this information, rationalizability is tested by checking whether there exists a split of prices (for public goods) or quantities (for private goods), such that the datasets of personalized prices and quantities for each household member satisfies GARP. This model, for an arbitrary number of goods, is first described by Cherchye et al. (2011); these authors also give a integer programming formulation. Talla Nobibon et al. (2013) provide a large number of practical and theoretical computational results for this problem. First, they prove it is NP-HARD. Furthermore, they describe a more compact integer programming formulation, and provide a simulated annealing based meta-heuristic. They compare the computational results of these different integer programming formulations and heuristics and find that the heuristic approach is capable of tackling larger datasets and seldom fails to find a feasible split if one exists. In contrast to the result in the general case, our results in Chapter 6 show that, even when dropping transitivity, the test remains NP-HARD.

1.4 Revealed Stochastic Preference

In the previous sections, we looked at methods that decide whether a set of observations can be rationalized by one or more decision makers, using different forms of utility functions, or different ways in which the decision making process can be split over different decision makers. However, we assumed that utility functions and preferences remained constant. As

a result, if a choice situation repeats itself, we expect that the decision maker chooses the same alternative. However, it is commonly observed in experiments on choice behaviour that if a person is given the same choice situation multiple times, the choice she makes changes. One possible way of explaining this behaviour is by stochastic preferences, as pioneered by Block and Marschak (1960). Theories of stochastic preferences state that, while at any point in time a decision maker has a preference ordering over all alternatives, these preferences are not constant over time. Observed behaviour is rationalizable by stochastic preferences, if and only if there exists a set of utility functions and a probability distribution over those utility functions, such that the frequency that an alternative is chosen in a choice situation is equal to the probability that this alternative has the highest utility in that situation. We also note that many results on stochastic preferences are for the case of finite choice sets, as opposed to the consumption setting, where there exist an infinite number of bundles that can be bought for a expenditure level and prices. For an overview, we refer to McFadden (2005).

A very general result was given by McFadden and Richter (1990), the *axiom of revealed stochastic preference* (ARSP), which gives a necessary and sufficient condition for rationalizability by stochastic preferences. The generality of this axiom allows it to be used for any form of choice situation, and all classes of decision rules. Besides the axiom, McFadden and Richter also provide a linear programming problem for which the existence of a solution is a necessary and sufficient condition for rationalizability. Both are not straightforward to operationalize, since ARSP places a condition on every possible subset of observations, which gives an exponential number of conditions in the number of observations. Furthermore, each condition requires finding a decision rule from all allowed decision rules which maximizes some function, which can in itself be an NP-HARD problem (for example when the class of decision rules being tested are decisions based on linear preference orders, as this means solving an NP-HARD linear ordering

problem (Karp, 1972)). The linear program on the other hand contains one variable for every possible decision rule within a class of decision rules, a number which is often exponential in the number of choice alternatives. We will now look at some rationalizability tests for specific classes of decision rules and specific choice settings.

Binary choice settings have attracted considerable attention within the literature on stochastic preferences. In such a setting, decision makers are faced with only two options in each choice situation. Block and Marschak (1960) work with this setting and search for conditions for rationalizability by stochastic strict linear preferences. Using the observed frequency with which alternatives are chosen over other alternatives, they find two simple classes of inequalities which are necessary conditions, and they prove sufficiency for datasets containing three choice alternatives. It was conjectured that these conditions are also sufficient for any number of choice alternatives, until a counter-example by Megiddo (1977) showed this was not the case for thirteen alternatives. Dridi (1980) closed the discussion by proving the conditions were sufficient for five alternatives and by providing a counterexample for six. Later, Suck (1992) showed that the necessary and sufficient conditions were equivalent to a membership test of the linear ordering polytope. This proves NP-HARDNESS of the rationalizability test, and thus that no polynomial size system of linear conditions exists unless P = NP. Research on the linear ordering polytope has provided a full facet description for up to seven alternatives (Martí and Reinelt, 2011), which already contains over 87,000 constraints. For eight alternatives, this rises to above 480 million. As a result, current tests restrict themselves to about 5 choice alternatives in the dataset (Regenwetter et al., 2011; Regenwetter and Davis-Stober, 2012). In Chapter 7, we propose using column generation on McFadden and Richter's linear program for testing rationalizability for datasets containing a larger number of choice alternatives.

We are aware of two classes of decision rules for which stochastic rational-

izability is easy to test. Davis-Stober (2012) provides a polynomial number of conditions which are necessary and sufficient for stochastic rationalizability by a simple heuristic decision rule. In Davis-Stober's setting, choice alternatives have two attributes. The levels of these two attributes are such that, if the attributes are ranked according to the level of the attributes, the ranking according to the first attribute is the reverse of the ranking by the second. When faced with a choice, decision makers first set a difference threshold for the first attribute. For each alternative, the level of the first attribute is compared to the maximum level of this attribute in the choice set. Only alternatives for which this difference is below the threshold are considered in the second step. In this second step, the alternative with the highest level of the second attribute is chosen. In Chapter 8, we give a polynomial number of conditions which are necessary and sufficient for single-peaked preferences, a special case of strict linear preferences. In this setting, all alternatives are ranked along an axis. A single-peaked preference order then has a peak, which is the most preferred alternative, and for any pair of alternatives on the same side of the peak, the alternative closest to the peak is always preferred to the one further away.

Returning to the setting of consumer purchases (and thus infinite choice sets), Bandyopadhyay et al. (1999) formulate the *weak axiom of stochastic revealed preference* (WARSP). This axiom provides a necessary condition for rationalizability by stochastic preferences. Analogue to WARP, WARSP compares pairs of choice situations. Since the condition placed on these pairs is easy to test, WARSP allows a polynomial time test. Heufer (2011) and Kawaguchi (2014) build further on this work. Heufer provides a sufficient condition for rationalizability in terms of stochastic preferences. Kawaguchi (2014) constructs the *strong axiom of revealed stochastic preference* (SARSP), a necessary condition for rationalizability by stochastic preferences. Both of these conditions seem difficult to test, requiring in the case of Heufer a feasible solution to a linear program with an exponential number of constraints and variables. Kawaguchi's SARSP likewise requires

checking an exponential number of inequalities. Despite these challenges, Kitamura and Stoye (2014) develop a test which can be used to test rationalizability by stochastic preferences on consumption data, though for relatively small datasets. A key element in their approach is discretizing the dataset, to return to a setting with a finite number of choice options.

1.5 Conclusion

In this chapter, we surveyed some areas where revealed preference theory has been applied. We looked at unitary, collective and stochastic models of choice behaviour, with a focus on the computational aspects of testing these models. In the remainder of this thesis, we will present work on models in all three of these groups. Chapter 2 presents a short note on a more efficient algorithm for testing GARP and we also derive a lower bound on the computation time for WARP, SARP and GARP. Next, we look at goodness-of-fit measures. In Chapter 3 we present complexity results for Afriat's, Varian's and Houtman and Maks' index, showing the latter two are hard to compute, while for Afriat's index we present a polynomial time algorithm. Chapter 4 then looks at the Money Pump Index, showing that computing the MPI of an 'average' (mean or median) violation is difficult, but that the minimum or maximum MPI can be computed efficiently. In the next two chapters, we look at collective models. Chapter 5 investigates the (non)-equivalence between collective versions of WARP and SARP. In Chapter 6, we look at the computational complexity of collective versions of WARP, showing that most of these are already hard for 2 decision makers. However, for the most general model, where goods are not known to be private or public, the test for 2 decision makers is still easy. The final two chapters of this thesis handle stochastic preferences. Chapter 7 presents a model of stochastic strict preferences orderings. Testing this model is NP-HARD, we present an algorithm based on column generation to test it. Finally, Chapter 8 presents a stochastic model, where preferences are restricted to single-peaked preferences and we show this model is easy to

test.

Chapter 2

Testing the Axioms of Revealed Preference

2.1 Introduction

In this chapter, we investigate the computational complexity of testing three well-known axioms of revealed preference: the *weak, strong* and *generalized axioms of revealed preference* (WARP, SARP and GARP). As far as we are aware, each known method for testing GARP relies on computing the transitive closure of a directed graph. A straightforward algorithm for computing the transitive closure runs in $O(n^3)$ time, with n the number of observations in the dataset. Algorithms based on matrix multiplication have better worst-case bounds of $O(n^\alpha)$, with $\alpha \approx 2.37$. The main contribution of this chapter is the description of an $O(n^2)$ time algorithm for testing GARP, based on computing strongly connected components, for testing GARP. Furthermore, we also argue that any algorithm for testing WARP, SARP or GARP will need at least $O(n \log n)$ number of operations.

This chapter is organized as follows. In Section 2.2, we describe a graph

This chapter is the result of a collaboration with Fabrice Talla Nobibon and Frits C.R. Spieksma. A paper corresponding to this chapter will appear in the Journal of Optimization Theory and Applications (Talla Nobibon et al., 2014).

representation of the revealed preference relations and WARP, SARP and GARP. In Section 2.4, we present an $O(n^2)$ time algorithm for testing GARP. Finally, in Section 2.5 we prove the lower bound on any algorithm for testing WARP, SARP or GARP and we conclude in Section 2.6.

2.2 Notation and Definitions

Consider a unitary household acting in an economy with m goods and suppose that we have observed n (non-negative) consumption quantity bundles $q_i \in \mathbb{R}^m_+$ with corresponding positive prices $p_i \in \mathbb{R}^m_{++}$, for $i = 1, \ldots, n$. We denote the set of observations by $S = \{(p_i, q_i) : i \in N\}$ ($N = \{1, \ldots, n\}$). A bundle q_i is revealed preferred over another bundle q_j, denoted by $q_i \; R_0 \; q_j$, if and only if $p_i q_i \geq p_i q_j$ (and $q_i \neq q_j$). For every dataset S, there exists an associated graph $G(S) = (V, A)$, defined as follows. For every observation i in the dataset, there exists a vertex $i \in V$. For every pair of vertices i, j, an arc $(i, j) \in A$ exists if there is a direct revealed preference relation $q_i \; R_0 \; q_j$ and $q_i \neq q_j$. The length of an arc, denoted by $\ell(i, j)$, is given by $p_i q_j - p_i q_i$. This means that the length of an arc is either exactly 0, or negative. We now define WARP, SARP or GARP in terms of this graph $G(S)$.

Definition 2.2.1. *A dataset S satisfies* WARP *if and only if the associated graph $G(S)$ contains no cycles consisting of 2 arcs.*

Definition 2.2.2. *A dataset S satisfies* SARP *if and only if the associated graph $G(S)$ contains no cycles.*

Definition 2.2.3. *A dataset S satisfies* GARP *if and only if the associated graph $G(S)$ contains no cycles of negative length.*

The equivalence with the definitions given in Chapter 1 is straightforward. A cycle consisting of 2 arcs will only exist if and only if there is a pair of observations such that $q_i \; R_0 \; q_j$ and $q_j \; R_0 \; q_i$, which is a violation of WARP. A cycle of any length is likewise only possible if there is a sequence

of bundles $i, j, \ldots, k$, such that $q_i \; R_0 \; q_j \; R_0 \; \ldots \; R_0 \; q_k \; R_0 \; q_i$, which is a violation of SARP. A cycle of negative length implies at least one of the direct revealed preference relations in the previous sequence is strict; a violation of GARP.

In this chapter, we are interested in the following decision problems.

Problem 2.2.1. *Testing* GARP *(*WARP*,* SARP*)*
Instance: A dataset S.
Question: Does S satisfy GARP*?* (WARP, SARP)

2.3 The current State-of-the-art concerning WARP, SARP and GARP

The question of testing whether a given dataset S satisfies WARP, SARP or GARP can be answered in polynomial time. We now give an informal sketch of the procedures used (see Varian (2006)). Using $G(S)$, WARP can be tested by checking for each pair of vertices $i, j \in V$, whether both $(i, j) \in A$ and $(j, i) \in A$; WARP is violated if and only this is the case. Because building the graph $G(S)$ can be done in $O(n^2)$, and we perform $O(n^2)$ comparisons in total, we infer that testing WARP can be done in $O(n^2)$ time. For testing SARP, the graph $G(S)$ is tested for *acyclicity*. Checking whether $G(S)$ is acyclic is done in time $O(n^2)$ using, for example, a topological ordering algorithm. Thus, testing SARP can be done in $O(n^2)$ time.

The current algorithm for testing GARP is based on computing the transitive closure of the graph $G(S)$ (Varian, 1982, 2006) and proceeds as follows. Given $G(S) = (V, A)$, its transitive closure is represented by the graph $G_C(S) = (V, A_C)$. The set of arcs A_C is constructed as follows, if there is an arc $(i, j) \in A$, then there is an arc $(i, j) \in A_C$, with length $\ell(i, j) = p_i q_j - p_i q_i$. If there is no arc $(v_0, v_t) \in A$, but there is a sequence of vertices $[v_0, v_1, \ldots, v_t]$, such that $(v_{i-1}, v_i) \in A$ for all $i = 1, \ldots, t$, i.e.,

there is a path from v_0 to v_t in $G(S)$, then there is an arc $(v_0, v_t) \in A_C$, with $\ell(v_0, v_t) = 0$. For this graph $G_C(S)$ it can then be tested whether there exists a pair of vertices $i, j \in V$, such that both $(i, j) \in A_C$ and $(j, i) \in A_C$ and either $\ell(i, j) < 0$, $\ell(j, i) < 0$ or both. The bottleneck of this procedure is the computation of the transitive closure graph $G_C(S)$. Varian (1982) uses the $O(n^3)$-algorithm proposed by Warshall (1962) and mentions the possibility of computing the transitive closure using matrix multiplication. In the literature, the best-known algorithms for matrix multiplication on general matrices runs in time $O(n^{2.376})$ (Coppersmith and Winograd, 1990), some recent papers improve this bound to $O(n^{2.373})$ (Williams, 2012; Gall, 2014).

2.4 The Algorithm

The algorithm we present in this section is based on the computation of *strongly connected components* of the graph $G(S) = (V, A)$. A strongly connected component of a graph is a maximal (sub)set of vertices $V_k \subseteq V$, such that for every pair of vertices $i, j \in V_k$, there is a path from i to j and vice versa (Tutte, 1961). Notice that for every pair of distinct strongly connected components V_k, V_l $(k \neq l)$, it is the case that $V_k \cap V_l = \varnothing$. Indeed, if this were not the case, there exists a vertex $i \in (V_k \cap V_l)$, and for every vertex $a \in V_k$ and every vertex $b \in V_l$, there is a path from a, through i, to b and a path from b, through i, to a. Thus, the set $V_k \cup V_l$ is a single strongly connected component. We now define the graph $G_{SCC}(S) = (V, A_{SCC})$ as follows. If there exist an arc $(i, j) \in A$, and vertices i and j are in the same strongly connected component, then $(i, j) \in A_{SCC}$. If i and j are in different strongly connected components, then $(i, j) \notin A_{SCC}$. We now state the following result.

Lemma 2.4.1. *A dataset S satisfies* GARP *if and only if, for every arc $(i, j) \in A_{SCC}$, $\ell(i, j) = 0$.*

Proof. $\Rightarrow$) Suppose that the dataset S satisfies GARP. In this case, the

graph $G(S)$ has no cycles of negative length. Now consider the graph $G_{SCC}(S)$. By construction, any arc $(i,j) \in A$, which is part of a cycle in $G(S)$, is included in A_{SCC}. This is straightforward, since if there is a cycle involving the arc (i,j), by definition there is a path from i to j and from j to i and i and j are in the same strongly connected component. Also, for any arc $(i,j) \in A$ which is not part of any cycle, $(i,j) \notin A_{SCC}$, since there is no path from j to i, and the two vertices are in different components. Given that for each arc $\ell(i,j) \leq 0$, the existence of any arc in a cycle with $\ell(i,j) < 0$ implies a cycle of negative length. Now suppose there is an arc $(i,j) \in A_{SCC}$ with $\ell(i,j) < 0$. By the previous arguments, there is a cycle of negative length in $G(S)$.

$\Leftarrow$) Now suppose that for every arc $(i,j) \in A_{SCC}$, $\ell(i,j) = 0$. By construction, for each arc $(k,l) \in A$ involved in a cycle in $G(S)$, it is also the case that $(k,l) \in A_{SCC}$. Since there is no arc $(i,j) \in A_{SCC}$ with $\ell(i,j) < 0$, this implies there are no arcs of negative length involved in any cycle in $G(S)$ and thus GARP is satisfied. □

We propose the following algorithm for testing GARP. In step 1, graph $G(S)$ is built from the dataset S. The second step involves computing the strongly connected components of $G(S)$. We will be basing our complexity result on Tarjan's algorithm (Tarjan, 1972), which achieves a strong worst-case bound and is relatively simple. This algorithm uses a depth-first search, sequentially labelling all vertices in a graph while following the arcs. When previously labelled nodes are encountered a cycle exists, and the algorithm works backwards towards a root node of the corresponding strongly connected component. Given the strongly connected components, we build the graph $G_{SCC}(S) = (V, A_{SCC})$. Given this graph and its set of arcs, we can test whether all arcs in A_{SCC} have a length of 0. The main difference between Algorithm 1 and the current procedure (see Varian (2006)) for testing GARP stems from the fact that the former algorithm computes the strongly connected components of a directed graph whereas the latter computes the transitive closure of a matrix.

Algorithm 1 An algorithm for testing GARP

1: Build the graph $G(S)$
2: Compute the strongly connected components of $G(S)$
3: Build the graph $G_{SCC}(S) = (V, A_{SCC})$
4: **if** every arc (i, j) present in A_{SCC} has $\ell(i, j) = 0$ **then** return GARP satisfied
5: **else** return GARP violated

Theorem 2.4.1. *Algorithm 1 tests* GARP *in time* $O(n^2)$.

Proof. The correctness of the algorithm results from Lemma 2.4.1. Let us now analyse its complexity. The first step of the algorithm, building the directed graph $G(S)$, can be done in time $O(n^2)$ because we check $p_i q_j - p_i q_i$ for every pair of observations. The second step, computing the strongly connected components, is completed in time $O(n^2)$ when implemented using Tarjan's algorithm (Tarjan, 1972). Given the strongly connected components, building $G_{SCC}(S) = (V, A_{SCC})$ also takes $O(n^2)$ time, since for every pair of nodes we test whether they are in the same strongly connected component. Finally, the last step (the **if** loop) has a running time of $O(n^2)$. Therefore, the overall running time of the algorithm is bounded from above by $4 \times O(n^2) = O(n^2)$. □

2.5 Lower Bounds

Of course, a valid question is whether the $O(n^2)$ algorithms for WARP, SARP and GARP can be sped up even further. We note that, in the discussion so far, the number of goods, m, is seen as fixed. However, if the number of goods is part of the input, our algorithm becomes a $O(n^2 m)$ method. Of course, in that case reading the data takes $O(nm)$ time, and can be seen as a lower bound for any algorithm testing WARP, SARP and GARP. In this section, we derive a lower bound for testing these problems that does not rely on the effort needed to read the data. We will show that, even when

the data are known implicitly, an $\Omega(n \log n)$ lower bound still applies. More precise, we derive a lower bound of $\Omega(n \log n)$ on testing WARP, SARP and GARP under an algebraic computation tree model of computation. This is achieved by using a reduction from the *Element Distinctness* problem (Ben-Or, 1983; Yao, 1991).

Problem 2.5.1. *Element Distinctness*
Instance: A set x_1, x_2, ..., x_k *of* k *positive integers.*
Question: Are the integers x_i, $i = 1, \ldots, k$, *pairwise distinct?*

Using a topological method, Yao (1991) proves that any algebraic computation tree that solves the k-Element Distinctness problem has a lower bound complexity of $\Omega(k \log k)$. We next show that this lower bound is valid for WARP, SARP and GARP by arguing that an algorithm for testing these can also be used for determining whether k integers are pairwise distinct.

Given an instance x_1, x_2, ..., x_k of the Element Distinctness problem, we build a dataset S as follows. We assume that there are k goods. To describe the prices and the quantities of all goods for each observation we make use of a 'default' price (quantity) for each good. The vector of default prices is $(x_1 - 0.1,\ x_2 - 0.1,\ \ldots,\ x_k - 0.1)$ whereas the vector of default quantities is $(0,\ 0,\ \ldots,\ 0)$. Notice that to describe these default vectors, we need $O(k)$ operations. We consider a dataset S with k observations where an observation is identified by the index of a good. This index means that for the considered observation, that particular good, let us say j, has the price of $x_j + 0.1$ (instead of $x_j - 0.1$ as in the default vector) whereas the quantity of that good is now 1 (instead of 0 in the default quantity vector). The price (respectively the quantity) of each remaining good is exactly its default price (respectively its default quantity). Notice that all the quantities in S are pairwise distinct. Also, observe that this second part of our reduction requires $O(k)$ operations because given the default price and quantity vectors, we need exactly $O(k)$ numbers to describe the dataset S. We now prove that the dataset S satisfies WARP, SARP or GARP

if and only if the considered instance of the Element distinctness problem is a yes instance.

Consider the directed graph $G(S)$ built from S and observe that there is an arc from i to j if and only if $p_i q_i \geq p_i q_j$; that is $x_i + 0.1 \geq x_j - 0.1$, as x_i and x_j are integers, this is equivalent to $x_i \geq x_j$. As a result, if there is a cycle $(s,u),(u,v),\ldots,(z,t)(t,s)$ then the two-cycles $(s,u)(u,s)$, $(u,v),(v,u)$, $\ldots$, and $(t,s)(s,t)$ are all present. This observation implies that if the graph does not contain any two-cycle then it does not contain any cycle and vice-versa. As a result, S satisfies WARP if and only if S satisfies SARP. Another remark is that for all pair of observations i and j we have $p_i q_i = x_i + 0.1 \neq x_j - 0.1 = p_i q_j$, because x_i and x_j are integers. This means that for any pair of observations i and j we have $p_i q_i \neq p_i q_j$. These inequalities imply that S satisfies GARP if and only if S satisfies SARP. Thus, for this particular dataset S, testing SARP, WARP or GARP is equivalent. Observe now that a two-cycle $(i,j),(j,i)$ is present in our graph if and only if $x_i = x_j$; in other words, there is a two-cycle in our graph if and only if the two elements x_i and x_j are identical. It immediately follows that the dataset S satisfies WARP, SARP and GARP if and only if the considered instance of the Element Distinctness problem is a yes-instance. This proves that any algorithm for solving WARP, SARP or GARP can be used to solve the Element Distinctness problem. Therefore, the lower bound of $\Omega(k \log k)$ for solving the Element Distinctness problem is directly applicable to any algorithm for solving WARP, SARP or GARP.

We formalize this result in the following theorem.

Theorem 2.5.1. *Any algorithm for testing either* WARP, SARP *or* GARP *on a dataset S with n observations has a running time bounded from below by $\Omega(n \log n)$.*

2.6 Conclusions

This chapter presents an $O(n^2)$ algorithm testing GARP, which improves upon the best known complexity for algorithms for testing GARP. Furthermore, we show that the element distinctness problem can be seen as a special case of testing WARP, SARP or GARP. As it has been proven that element distinctness cannot be tested in less than $\Omega(n \log n)$ time, this provides a lower bound of the computational complexity of tests for these axioms.

Chapter 3

Goodness-of-fit Measures for Revealed Preference Tests

3.1 Introduction

As we noted in Chapter 1, tests of the axioms of revealed preference are 'sharp' tests: they only tell us whether or not observed behaviour is consistent with the revealed preference axiom that is being tested. When the dataset does not pass the test, there is no indication concerning the severity or the number of violations. To deal with this, a number of goodness-of-fit measures have been proposed in the literature to express how close a dataset is to satisfying rationality. In this chapter, we will focus on some classical goodness-of-fit measures, which are often used in applied work. In particular, we look at Afriat's index (AI) (Afriat, 1973), Houtman and Maks' index (HI) (Houtman and Maks, 1985) and Varian's index (VI) (Varian, 1990). These goodness-of-fit measures can be computed for different

This chapter is the result of a collaboration with Laurens Cherchye, Bram De Rock and Frits C.R. Spieksma. An article based on parts of this chapter has been published in Transactions on Economics and Computation (Smeulders et al., 2014b). We gratefully acknowledge helpful discussions with Gerhard Woeginger and Fabrice Talla Nobibon and interesting comments by the referees.

axioms of revealed preference. In the previous chapters, we looked at the weak, strong and general axioms of revealed preferences (WARP, SARP and GARP). In this Chapter, we will also consider the *homothetic axiom of revealed preference* (HARP). Section 3.2 provides a precise description of this axiom and the different goodness-of-fit measures.

These revealed preference axioms and goodness-of-fit measures have been used intensively in the literature. The first tests of the axioms of revealed preference go back to the sixties and seventies. Aggregated household consumption data was used in tests of SARP by Koo (1963, 1971), Koo and Hasenkamp (1972), Mossin (1972) and Landsburg (1981). Varian (1982) tested GARP using similar data. Only Koo tried to measure the severity of the rejections by focusing on the number of violations and using a measure similar to HI. Over the last decade, the goodness-of-fit measures have been used more and more often. Sippel (1997) tests relaxations of WARP, SARP and GARP related to AI. AI and GARP are used in papers by Mattei (2000), Harbaugh et al. (2001), Andreoni and Miller (2002), Février and Visser (2004), Choi et al. (2007, 2014), Dean and Martin (2010) and Burghart et al. (2013); the last four papers also use HI. VI and GARP appears in Cox (1997), Mattei (2000), Choi et al. (2007, 2014) and Dean and Martin (2010). For WARP, all three indices appear in Choi et al. (2007). To the best of our knowledge, there do not exist any studies that compute goodness-of-fit measures for HARP, although there exist papers in which HARP is tested (see for example Manser and McDonald (1988)).

It is generally thought that calculating AI is easy. However, to our knowledge, no exact algorithm is described in the literature. Varian (1990) provides an approximation algorithm, which comes within an additive error of $\left(\frac{1}{2}\right)^m$ of the true index-value in m GARP tests. As for the other two indices (HI and VI), it has been empirically recognized that computing them is computationally intensive. For instance, Varian (1990) writes:

> "Computing the set of efficiency indices [VI] that are as close as possible to 1 in some norm is substantially more difficult ...

> This approach is significantly more difficult from a computational perspective."

Similarly, Choi et al. (2014)[1] state:

> "All indices [VI and HI] are computationally intensive for even moderately large datasets."

In fact, because of the apparent difficulty to exactly calculate VI, some authors have focused on designing approximate heuristics, see, for example Varian (1993), Tsur (1989) and Alcantud et al. (2010).

The goal of the current chapter is to give a theoretical foundation for these practical observations and to strengthen the existing results. As far as we are aware, explicit complexity results are known only for index HI. More specifically, Houtman and Maks establish a link between their index for SARP and feedback vertex sets on a digraph, which implies NP-Hardness. Next, Dean and Martin (2010) state that HI for GARP is also NP-hard.

We establish the computational complexity for every combination of the three goodness-of-fit measures (AI, VI and HI) and the four revealed preference axioms (GARP, SARP, WARP and HARP) mentioned above. We will refer to these problems as {A,V,H}I-{G,S,W,H}ARP, where choosing a symbol from the set {A,V,H} and a symbol from the set {G,S,W,H} identifies the particular problem. For example, AI-GARP is the problem of computing the maximum index AI such that the dataset satisfies a relaxation of GARP. Our main results are summarized in Table 1, where a column corresponds to a specific axiom and a row to a specific measure, and where n stands for the number of observations. *'Inapproximable'* stands for: no polynomial-time algorithm can achieve a constant-factor approximation unless P = NP.

[1] This quote is found in a working paper version (January 2011) of this article.

	AI (sec 3.6)	VI (sec 3.4)	HI (sec 3.5)
WARP	n^2	Inapproximable	Inapproximable
SARP	$n^2 \log n$	Inapproximable	Inapproximable
GARP	$n^2 \log n$	Inapproximable	Inapproximable
HARP	n^3	Inapproximable	Inapproximable

Table 3.1: *Overview of Results*

The rest of this chapter unfolds as follows. The next section sets the stage by introducing HARP and the considered goodness-of-fit measures in more detail. Section 3.3 provides a statement of the computational problems we focus on. Sections 3.4 and 3.5 present our results on the computational complexity for the indices VI and HI. Section 3.6 contains a polynomial time algorithm for AI. Section 3.7 concludes.

3.2 Revealed preference concepts

We start by stating the homothetic axiom of revealed preference in 3.2.1. For the exact definitions of the other axioms of revealed preference, we refer to Chapter 1. Subsequently, we present the different goodness-of-fit measures in 3.2.2.

3.2.1 The Homothetic Axiom of Revealed Preference

As before, our analysis starts from a dataset $S = \{(p_i, q_i) | i \in N\}$, $(N = \{1, \ldots, n\})$ where $p_i \in \mathbb{R}^m_{++}$ and $q_i \in \mathbb{R}^m_{+}$ correspond to observations $i = 1, \ldots, n$. In this chapter, we will assume, without loss of generality, that prices are normalized such that $p_i q_i = 1$ for every observation i. Using this normalization, Varian (1983) gives the following definition of HARP.

Definition 3.2.1. *A normalized dataset S satisfies* HARP *if and only if for every sequence of observations, $i, j, k, \ldots, l$: $(p_i q_j)(p_j q_k) \ldots (p_l q_i) \geq 1$.*

This definition can be rewritten in the following way, which we will use throughout this chapter.

Definition 3.2.2. *A normalized dataset S satisfies* HARP *if and only if, for every sequence of observations, $i, j, k, \ldots, l$:* $\log(p_i q_j) + \log(p_j q_k) + \ldots + \log(p_l q_i) \geq 0$.

As is the case with WARP, SARP and GARP, there also exists a graph representation of HARP, which is convenient to test the axiom. We will refer to this graph as $G^H(S) = (V, A^H)$. In this graph, nodes are defined in the same way as for the graph $G(S)$, i.e., there is a node for every observation. However, unlike the graph $G(S)$, there exists an arc $(i, j) \in A^H$ and $(j, i) \in A^H$ for every pair of nodes $i, j \in V$, i.e. the graph $G^H(S)$ is complete. The length of an arc (i, j) is equal to $\log p_i q_j$ for each $(i, j) \in A^H$. Note that unlike the graph $G(S)$ used for the other axioms of revealed preference, the length of arcs can be strictly positive in the graph $G^H(S)$. In terms of this graph, HARP can be defined as follows.

Definition 3.2.3. *A normalized dataset S satisfies* HARP *if and only if the associated graph $G^H(S)$ contains no cycles of negative length.*

The main differences between the alternative axioms we consider in this chapter can be summarized as follows (see Varian (2006) for a more extensive discussion on the meaning of the axioms). Data consistency with WARP implies that the direct revealed preference relation R_0 is asymmetric. By construction this direct revealed preference relation is not transitive since $q_i R_0 q_j$ and $q_j \ R_0 \ q_k$ does not need to imply $q_i \ R_0 \ q_k$. This is no longer the case for the revealed preference relation R, which is the transitive closure of R_0. Data consistency with WARP is a necessary condition for data consistency with SARP and implies that the revealed preference relation is asymmetric (and transitive).

Next, data consistency with SARP means that consumption behaviour can be described as maximizing a utility function that generates single-valued

demand. Similarly, data consistency with GARP means that consumption behaviour can be described as maximizing a utility function that generates multi-valued demand. As such, GARP is a generalization of SARP and data consistency with GARP still makes the revealed preference relation R an asymmetric and transitive relation. Finally, data consistency with HARP means that consumption behaviour can be described as maximizing a utility function that is homothetic. This implies that GARP is a necessary condition for HARP and that the revealed preference relation is asymmetric and transitive. Figure 3.1 illustrates the relations between the different axioms of revealed preference.

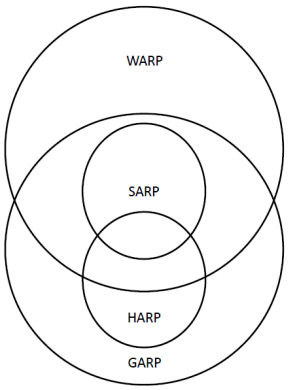


Figure 3.1: *Relations of the axioms of revealed preference*

3.2.2 Goodness-of-Fit Measures

In practice, direct application of any of the above revealed preference axioms to some given dataset effectively obtains a 'sharp' test: a dataset either satisfies the axiom or it does not. In other words, such a test allows us to conclude whether or not observed behaviour is 'exactly' consistent with the hypothesis of utility maximization (of a particular form, depending on whether we consider WARP, SARP, GARP or HARP). However, a dataset that is not exactly consistent may actually be very close to consis-

tency. For example, there may be only a limited number of observations that cause the observed violations of the axiom that is subject to testing. Or, the violations may be very insignificant in that small adjustments of the observations' expenditures (i.e. prices times quantities) may suffice to obtain consistency. Generally, it is interesting to quantify the degree to which a given dataset is close to consistency (see Varian (1990) for extensive motivation).

To account for these considerations, a number of goodness-of-fit measures have been described in the literature. Three often used measures are Afriat's efficiency index (AI), Varian's efficiency vector index (VI) and the Houtman and Maks index (HI). Essentially, the indices AI and VI look for minimal expenditure perturbations to obtain consistency with the revealed preference axiom under evaluation: the AI index applies a common perturbation to all observations, while the VI index allows a different perturbation for each individual observation. Next, the index HI identifies the largest subset of observations that are consistent with the axiom. Essentially, this quantifies the degree of violation in terms of the number of observations that are involved in a violation of the revealed preference axiom that is tested. We refer to Varian (2006) for a more detailed discussion of the different goodness-of-fit measures we evaluate.

To formally introduce our goodness-of-fit measures, we make use of the vector $e = (e_1, e_2, \ldots, e_n)$, with $0 \leq e_i \leq 1$. This vector introduces an index e_i for each observation $i \in N$, which relaxes the revealed preference relations R_0 and P_0 as follows:

$$\text{if } e_i(= e_i p_i q_i) \geq p_i q_j \text{ then } q_i R_0(e) q_j,$$

$$\text{if } e_i(= e_i p_i q_i) > p_i q_j \text{ then } q_i P_0(e) q_j.$$

$R(e)$ and $P(e)$ represent the transitive closures of $R_0(e)$ and $P_0(e)$. These newly defined relations $R_0(e)$, $P_0(e)$, $R(e)$ and $P(e)$ give rise to relaxed versions of the earlier axioms of revealed preference, which are defined for a given vector e. Clearly these axioms comply with the original versions of WARP, SARP, GARP and HARP as soon as $e_i = 1$ for all $i \in N$.

Definition 3.2.4. *For a given* $e = (e_1, e_2, \ldots, e_n)$*, with* $0 \leq e_i \leq 1$*, a normalized dataset* S *satisfies*

- WARP*(e): If and only if, for each pair of distinct bundles,* q_i, q_j*: if* $q_i \; R_0(e) \; q_j$ *then it is not the case that* $q_j \; R_0(e) \; q_i$*.*
- SARP*(e): If and only if, for each pair of distinct bundles,* q_i, q_j*: if* $q_i \; R(e) \; q_j$ *then it is not the case that* $q_j \; R_0(e) \; q_i$*.*
- GARP*(e): If and only if, for each pair of distinct bundles,* q_i, q_j*: if* $q_i \; R(e) \; q_j$ *then it is not the case that* $q_j \; P_0(e) \; q_i$*.*
- HARP*(e): If and only if, for every sequence of observations,* $i, j, k, \ldots, l (= 1, \ldots, n)$*:* $\log(p_i q_j) + \log(p_j q_k) + \ldots + \log(p_l q_i) \geq \log(e_i) + \log(e_j) + \ldots + \log(e_l)$*.*

To define the Afriat Index (AI), we assume that $e_1 = \cdots = e_n$, which does indeed comply with a common perturbation for all observations. The index AI equals the supremum over all values for which the data is consistent with the tested revealed preference axiom. More precisely, if $e = 1$, then the data is consistent with the tested axiom, while if $e < 1$, then this indicates that we need to pertubate the data to make it consistent with the revealed preference axiom under study. The smaller the number e is, the higher the perturbation or, alternatively, the more severe the rejection of the axiom. Finally, we notice that e is well-defined. If for a given e the data is consistent with, for example, WARP(e), then the same holds for all $e' < e$. Indeed, by construction we have that the revealed preference relations in terms of e' are always a subset of the ones in terms of e (e.g. $R_0(e') \subseteq R_0(e)$).

The Varian Index (VI) differs from the index AI by allowing for observation specific perturbations. The index VI equals the vector e that is closest to 1, for some given norm, such that the data satisfies the revealed preference axiom under study. For example, if we use the quadratic norm, then VI

should minimize $\sum_i (1 - e_i)^2$ such that, for example, WARP(e) is satisfied. Further, the index VI is subject to the same qualifications as the index AI.

Finally, the Houtman and Maks index (HI) equals the size of the largest subset of observations which satisfy the axioms of revealed preference. Formally, this complies with restricting the possible values of e_i so that $e_i \in \{0, 1\}$.

Graph representation

In order to verify whether a dataset actually satisfies some revealed preference axiom for a given vector e, it is natural to adapt the graphs $G(S)$ and $G^H(S)$. For a given dataset S and vector e, we construct the associated graph $G_e(S)$ as follows. In this graph, there is a node for every observation. Next, for each pair of observations i, j $(q_i \neq q_j)$, there is an arc from node i to node j when $e_i \geq p_i q_j$. The length of this arc is equal to $p_i q_j - e_i$. The graph $G_e(S)$ will be used to test WARP, SARP and GARP. To test HARP, we make use of another graph $G_e^H(S)$. In this graph, nodes are defined in the same way as for the graph $G_e(S)$, i.e., there is a node for every observation. Furthermore, for every pair of nodes (i, j), there exists an arc from i to j and an arc from j to i. The lengths of these arcs are given by $\log(p_i q_j) - \log(e_i)$.

The axioms of revealed preference can then be formulated as follows:

Definition 3.2.5. *For a given $e = (e_1, e_2, \ldots, e_n)$, with $0 \leq e_i \leq 1$, the dataset S satisfies*

- WARP*(e): If and only if each cycle consisting of 2 arcs in the graph $G_e(S)$ involves observations that have identical bundles.*
- SARP*(e): If and only if each cycle in the graph $G_e(S)$ contains only observations with identical bundles.*

- GARP*(e): If and only if the graph $G_e(S)$ does not contain any cycles of negative length.*
- HARP*(e): If and only if the graph $G_e^H(S)$ does not contain any cycles of negative length.*

3.3 Problem statement

In this section we introduce the tools that we need to prove the results announced in Table 3.1. Using the graph representation described in Section 3.2.2, we are now in a position to define an optimization problem that measures how close a given dataset is to satisfying a particular axiom of revealed preference. This leads to twelve different optimization problems. For example, for SARP(e) we obtain the problems AI-SARP, VI-SARP and HI-SARP, each corresponding to a specific index. Straightforward adaptations define the problems AI-{W,G,H}ARP, VI-{W,G,H}ARP and HI-{W,G,H}ARP. For compactness, we only state the optimization problems with respect to SARP; the optimization problems corresponding to {W,G,H}-ARP are defined analogously.

Problem 3.3.1. VI-SARP
Given a dataset S, maximize $\sum_{i=1}^{n} e_i$, with $0 \leq e_i \leq 1$ for each i, while S satisfies SARP*(e).*

Clearly, objective functions other than $\sum_{i=1}^{n} e_i$ are possible. We will come back to this issue in Section 3.4.

Problem 3.3.2. HI-SARP
Given a dataset S, maximize $\sum_{i=1}^{n} e_i$, with $e_i \in \{0, 1\}$ for each i, while S satisfies SARP*(e).*

Results concerning this problem will be given in Section 3.5.

Problem 3.3.3. AI-SARP
Given a dataset S, maximize e, with $0 \leq e \leq 1$, while S satisfies SARP*(e).*

3.4 Varian's Index

Clearly, when given a vector $e = (e_1, \ldots, e_n)$, there are different ways to specify an objective function measuring the quality of e. Obvious candidates are minimize $\sum_{i=1}^{n}(1-e_i)$, minimize $\sum_{i=1}^{n}(1-e_i)^2$ or minimize $\max_i\ (1-e_i)$. In fact, all these objective functions can be captured by considering minimize $\sum_{i=1}^{n}(1-e_i)^\rho$ for $\rho \geq 1$. Observe that, since *minimize* $\lim_{\rho\to\infty}\sum_{i=1}^{n}(1-e_i)^\rho$ is equivalent to *minimize* $\max_i\ (1-e_i)$, the Afriat index arises when $\rho \to \infty$. The results in this section are phrased for $\rho = 1$, i.e., for the case where we minimize $\sum_{i=1}^{n}(1-e_i)$ or equivalently maximize $\sum_{i=1}^{n} e_i$. At the end of the section we point out that the reduction remains valid for every fixed $\rho \geq 1$. Notice that for every fixed ρ the problem is hard, while for $\rho \to \infty$ the problem becomes easy.

Let us now consider the following decision problem associated with VI-SARP (VI-SARP$_d$):

Problem 3.4.1. VI-SARP$_d$
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $0 \leq e_i \leq 1$, such that

(i) The dataset S satisfies SARP*(e), and*

(ii) $\sum_{i=1}^{n} e_i \geq Z$?

Obviously, being able to solve Problem 3.3.1 from Section 3.3, i.e., being able to solve VI-SARP, in polynomial time implies that VI-SARP$_d$ can be solved in polynomial time. That, however, is unlikely, as witnessed by our next result:

Theorem 3.4.1. VI-SARP$_d$ *is NP-complete.*

Proof. First, we show that the VI-SARP$_d$ is in NP, i.e., we show that for every Yes-Instance, there is a 'short' certificate. Suppose an instance of VI-SARP$_d$ is a Yes-Instance, and $(e_1, e_2, \ldots, e_n)$ is a solution, then there

exists a solution $(e'_1, e'_2, \ldots, e'_n)$ which can be described as a function of the p and q vectors and Z. Indeed, since an e_i value basically determines the presence or absence of arcs emanating from the node corresponding to observation i, only $O(n)$ breakpoint values of e_i are relevant (and these can be described as $p_i q_j / p_i q_i$). Thus if $(e_1, e_2, \ldots, e_n)$ is a solution, we can increase an individual e_i value until it meets a breakpoint value (bp_i). At this breakpoint value, SARP may no longer be satisfied, since the newly appearing arc(s) can lead to a violation. However, for any value of e_i below this breakpoint, no new arcs appear and SARP is satisfied. We set $e'_i = bp_i - (\sum_{k=1}^{n} bp_k - Z)/n$, for all $i \in N$. Notice that $(e'_1, e'_2, \ldots, e'_n)$ is feasible, since

$$\sum_{i=1}^{n} e'_i = \sum_{i=1}^{n} (bp_i - (\sum_{k=1}^{n} (bp_i) - Z)/n) = Z. \tag{3.1}$$

Also notice that $(e'_1, e'_2, \ldots, e'_n)$ is a short certificate, which can be completely described in a polynomial number of bits in regards to the inputs, p, q and Z. Given this certificate, calculating $\sum_{i=1}^{n} e'_i$, building the graph $G_{e'}(S)$ and testing it for acyclicity can be done in polynomial time, thus the problem is in NP.

Next, we prove that VI-SARP$_d$ is NP-hard by a reduction from the well-known NP-hard independent set problem Karp (1972), which is formulated as follows:

Problem 3.4.2. INDEPENDENT SET
Instance: A graph $G = (V, E)$ and a number k.
Question: Does there exist a subset $V' \subseteq V$ of at least k vertices, such that for every pair of vertices $i, j \in V'$, the edge (i, j) is not in E?

Given an instance of IS we now construct the following instance of VI-SARP$_d$. For every node $i \in V$, there is an observation in VI-SARP$_d$: $n := |V|$. The vectors $p_i = (p_{i,1}, \ldots, p_{i,n}), q_i = (q_{i,1}, \ldots, q_{i,n})$ are created as follows.

We set, for $i = 1, \ldots, n$, $q_{i,i} := 1$, all remaining $q_{i,j} := 0$. Further, we set $p_{i,i} := 1$, for $i = 1, \ldots, n$. If there is an edge between node i and node j in G, i.e., if $\{i, j\} \in E$, then $p_{i,j} := \epsilon$ (for some $0 < \epsilon < \frac{1}{n}$), otherwise $p_{i,j} := 2$. Finally, we set $Z := k$. This completes the description of the instance of VI-SARP$_d$. Notice that this construction implies that if an edge exists between i and j in G, then $p_i q_j = p_j q_i = \epsilon$, else $p_i q_j = p_j q_i = 2$.

We now argue the equivalence between IS and VI-SARP$_d$. Suppose the instance of independent set is a yes-instance, i.e., an independent set of size at least k exists. For every vertex in that independent set, set $e_i = 1$ and for every other vertex set $e_i = 0$. It is clear that $\sum e_i \geq Z$. Consider the graph $G_e(S)$, and recall that an arc is present from i to j if and only if $p_i q_j \leq e_i$. We claim that the graph $G_e(S)$ is acyclic. Indeed, notice that vertices outside the independent set will not have any outgoing arcs in $G_e(S)$ since for each such vertex i: $p_i q_j - e_i = p_i q_j > 0$. Also notice that no arc connects two observations corresponding to nodes in the independent set, since for a pair of such observations i, j we have $p_i q_j - e_i = p_j q_i - e_j = 2 - 1 > 0$. Thus, arcs in $G_e(S)$ only exist from vertices in the independent set to vertices outside the independent set. It follows that the graph is acyclic.

Now, suppose that the instance of VI-SARP$_d$ is a yes-instance, so $\sum e_i \geq Z = k$. Then for at least k observations $e_i > \epsilon$; if not, at most $k - 1$ e_i-values exceed ϵ; since $e_i \leq 1$, $\sum e_i$ is then bounded by $k-1+(n-k+1)\epsilon < k - 1 + 1 = k$, which contradicts the requirements for a yes-instance. We will call such an e_i value *large*. We claim that the vertices with large e_i-values constitute an independent set in G. Indeed, consider two vertices i and j with large values e_i and e_j. If i and j are connected in G, then $p_i q_j = p_j q_i = \epsilon$, implying that there is an arc in the graph $G_e(S)$ from i to j and from j to i, which is a cycle. Therefore i and j are not connected in G. Thus, the set of vertices with large e_i is an independent set of size at least k. □

We now proceed with VI-GARP; its corresponding decision problem is VI-GARP$_d$:

Problem 3.4.3. VI-GARP$_d$
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $0 \leq e_i \leq 1$, such that

(*i*) *The dataset S satisfies* GARP*(e), and*

(*ii*) $\sum_{i=1}^{n} e_i \geq Z$*?*

Theorem 3.4.2. VI-GARP$_d$ *is NP-complete.*

Proof. This proof is an adaptation of the proof of Theorem 3.4.1. Observe that we are now interested in the question whether there exists a cycle in the graph $G_e(S)$ that has negative length (see Definition 3.2.5).

Again, VI-GARP$_d$ is easily seen to be in NP. The certificate is the same, calculating $\sum_{i=1}^{n} e_i$, building the graph $G_e(S)$ and testing Definition 3.2.5 can all be done in polynomial time.

The instance that we build using independent set is exactly the same as in the proof of Theorem 3.4.1. We now show the equivalence. From the proof of Theorem 3.4.1, we know that if an independent set of size k exists, we can find a vector e for which $\sum e_i \geq Z = k$ and SARP(e) is satisfied. As GARP(e) is a relaxation of SARP(e), GARP(e) holds as well.

Vice versa, we now argue that a yes-instance of VI-GARP$_d$ corresponds with an independent set of size at least k. Consider two nodes in G, i and j, and assume that both e_i and e_j are large. If i and j are connected in G, then an arc from i to j, and an arc from j to i, both with negative length are present in $G_e(S)$. This however, is impossible since the instance of

VI-GARP$_d$ is a yes-instance. Thus, since there are at least k observations with a large e-value, an independent set of size at least k exists in G. □

Next we consider the problem VI-WARP and its corresponding decision problem VI-WARP$_d$:

Problem 3.4.4. VI-WARP$_d$
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $0 \leq e_i \leq 1$, such that

(*i*) *The dataset S satisfies* WARP*(e), and*

(*ii*) $\sum_{i=1}^{n} e_i \geq Z$*?*

Theorem 3.4.3. VI-WARP$_d$ *is NP-complete.*

Proof. For VI-WARP$_d$, the proof of Theorem 3.4.1 is also easily adapted. Again no changes are made to the graph construction, the only difference compared to VI-SARP$_d$ is that cycles are now allowed, as long as they involve more than 2 vertices (see Definition 3.2.5). Clearly, VI-WARP$_d$ is in NP.

From the proof of Theorem 3.4.1, we know that if an independent set of size k exists, we can find a vector e for which $\sum e_i \geq Z = k$ and the graph $G_e(S)$ is acyclic. As it is acyclic, clearly no cycles involving only two vertices exist and WARP(e) is satisfied.

Finally, if the instance of VI-WARP$_d$ is a yes-instance, we claim that an independent set of size k exists. As shown before, the observations for which the e_i value is large are not connected in the graph G, so they form an independent set. □

We end this section with the problem VI-HARP; here is the corresponding decision version:

Problem 3.4.5. VI-HARP$_d$
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $0 \leq e_i \leq 1$, such that

(*i*) *The dataset S satisfies* HARP*(e), and*

(*ii*) $\sum_{i=1}^{n} e_i \geq Z$*?*

Theorem 3.4.4. VI-HARP$_d$ *is NP-complete.*

Proof. As with the other axioms, VI-HARP$_d$ is in NP since the certificate consists of the e_i values. Testing the graph for Definition 3.2.5 can be done using a minimum cycle mean algorithm. This algorithm identifies the cycle for which the mean weight of its arcs is the minimum mean weight of all cycles in the graph. If the mean weight of the arcs of this cycle is positive, HARP(e) is satisfied.

An instance of VI-HARP$_d$ is built in the same way as the instances in the proofs of the previous Theorems. Consider now the equivalence. If an independent set of size k exists in G, we choose $e_i = 1$ for observations corresponding to nodes in the independent set and $e_i = \epsilon^{n+1}$ for the other nodes. Observe that the resulting graph $G_e^H(S)$ has the following properties:

(*i*) an arc in $G_e^H(S)$ emanating from an observation corresponding to a node not in the independent set has a length of either $\log \epsilon - \log \epsilon^{n+1}$, if there exists an edge between both nodes in G or $\log 2 - \log \epsilon^{n+1}$ otherwise.

(*ii*) an arc in $G_e^H(S)$ emanating from an observation corresponding to a node in the independent set has a length of either $\log \epsilon - \log 1$, if there exists an edge between both nodes in G or $\log 2 - \log 1$ otherwise.

Any cycle that contains only observations corresponding to nodes within the independent set has positive length, since the length of each arc equals $\log 2$. Further, a cycle in $G_e^H(S)$ going through an observation corresponding to a node not in the independent set contains an arc with length at least $\log \epsilon - \log \epsilon^{n+1}$. Hence, the length of this cycle is at least $\log \epsilon - \log \epsilon^{n+1} + n \log \epsilon = (n+1) \log \epsilon - \log \epsilon^{n+1} = 0$. Thus each cycle has non-negative length and the instance is a yes-instance of VI-HARP$_d$.

Consider now a yes-instance of VI-HARP$_d$. Clearly, there will be at least k observations with a large e_i value. Consider two nodes in G, each corresponding to an observation with a large e-value. If these two nodes are connected in G arcs of length $\log \epsilon - \log e_i$ and $\log \epsilon - \log e_j$ are present, yielding a negative cycle. Thus, two observations with large e-values can not correspond to nodes that are connected in G. □

From the NP-completeness of the decisions problems, it follows that the optimization problems VI-{W,S,G,H}ARP are NP-hard. Next, we show that not only is computing Varian's Index NP-hard, but also that approximating it in polynomial time is difficult, unless P = NP.

Theorem 3.4.5. *For each fixed $\delta > 0$, the existence of a polynomial time approximation algorithm* VI-{W,S,G,H}ARP *achieving a ratio of $O(n^{1-\delta})$ implies* P = NP.

Proof. Consider an instance of Independent Set, and the corresponding instance of VI-{W,S,G,H}ARP as constructed in Theorem 3.4.1. Clearly, if the optimum value of the VI-{W,S,G,H}ARP instance equals z, then the optimum value for the IS instance equals $\lfloor z \rfloor$ (if not, then then there exists an independent set of size $\lfloor z \rfloor + 1$, and by the previous reduction, we can find e so that $\sum_{i=1}^{n} e_i \geq \lfloor z \rfloor + 1$). Now, let z represent the optimum value of the instance of VI-{W,S,G,H}ARP, and let us assume that we have a polynomial time approximation algorithm for VI-{W,S,G,H}ARP achieving

a ratio $O(n^{1-\delta})$. Thus, more concrete, assume that we can compute in polynomial time a vector index e such that $\sum_{i=1}^{n} e_i \geq \frac{2z}{n^{1-\delta}}$. Given this vector-index we can find an independent set of size $max(1, \lfloor \frac{2z}{n^{1-\delta}} \rfloor)$. Since $n^{1-\delta} \times max(1, \lfloor \frac{2z}{n^{1-\delta}} \rfloor) \geq n^{1-\delta} \times \lfloor \frac{2z}{n^{1-\delta}} \rfloor \geq n^{1-\delta} \times \frac{\lfloor 2z \rfloor}{n^{1-\delta}+1} \geq n^{1-\delta} \times \frac{\lfloor 2z \rfloor}{2n^{1-\delta}} \geq \frac{\lfloor 2z \rfloor}{2} \geq \lfloor z \rfloor$, we conclude:

$$\max(1, \lfloor \frac{2z}{n^{1-\delta}} \rfloor) \geq \frac{\lfloor z \rfloor}{n^{1-\delta}}$$

It follows that we have a polynomial time algorithm achieving a ratio $n^{1-\delta}$ for IS. Håstad (1999) and Zuckerman (2006), have shown that an approximation algorithm for IS that runs in polynomial time can not guarantee a ratio of $n^{1-\delta}$, unless P = NP. Our result follows. □

Let us now return to the general objective function $\sum_{i=1}^{n} (1 - e_i)^{\rho}$ (with $\rho \geq 1$) given at the start of this section. We now consider the following problem:

Problem 3.4.6.
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $0 \leq e_i \leq 1$, such that

(*i*) *The dataset S satisfies* SARP*(e), and*

(*ii*) $\sum_{i=1}^{n} (1 - e_i)^{\rho} \leq Z$*?*

Theorem 3.4.6. *Computing Varian's Index is NP-hard for objective functions of the form $\sum_{i=1}^{n} (1 - e_i)^{\rho}$, for any fixed $\rho \geq 1$.*

Proof. Given an instance of Independent Set, create an instance of VI-{W,S,G,H}ARP as in the proof of their respective theorems with the following differences. Set $Z := n - k$ and let $0 < \epsilon < 1 - (\frac{n-k}{n-k+1})^{(1/\rho)}$. It can be easily checked that the equivalence holds. □

3.5 Houtman and Maks' Index

In this section, we consider the problems HI-{W,S,G,H}ARP. We give the problem HI-SARP, all other problems are analogous, differing only in the axiom of revealed preference to be satisfied. Notice that, in their original paper, Houtman and Maks (1985) already showed HI-SARP is an NP-COMPLETE problem, through a reduction from feedback vertex set, see also Dean and Martin (2010).

Problem 3.5.1. HI-{W,S,G,H}ARP$_d$
Instance: A dataset S and a number Z.
Question: Do there exist n numbers e_i, with $e_i \in \{0, 1\}$, such that

(*i*) *The dataset S satisfies* {W,S,G,H}ARP*(e), and*

(*ii*) $\sum_{i=1}^{n} e_i \geq Z$*?*

Theorem 3.5.1. HI-{W,S,G,H}ARP$_d$ *is NP-complete.*

Proof. Note that this result has been proven for HI-SARP$_d$ by Houtman and Maks (1985). For the other axioms of revealed preference, the proofs of NP-completeness for VI are easily extended to HI. As the choice of e_i is now limited to either zero or one, it is clear that every large $e_i = 1$ and every other $e_i = 0$, $i \in N$. □

The NP-hardness of the optimization problems follows from the NP-completeness of the decision problems.

Theorem 3.5.2. *For each fixed $\delta > 0$, the existence of a polynomial time approximation algorithm* HI-{W,S,G,H} *achieving a ratio of $n^{1-\delta}$ implies* P = NP.

Proof. Consider an instance of Independent Set, and the corresponding instance of HI-SARP as constructed in Theorem 3.4.1. Clearly, if the optimum value of the HI-SARP instance equals z, then the optimum value for the IS instance also equals z (if not, then then there exists an independent set of size $z+1$, and by the previous reduction, we can find e so that $\sum_{i=1}^{n} e_i \geq z+1$). Now assume that there exists a polynomial time approximation algorithm achieving a ratio of $n^{1-\delta}$ for HI-SARP, then we can find a vector-index so that $\sum_{i=1}^{n} e_i \geq \frac{z}{n^{1-\delta}}$. Given this vector-index we can find an independent set of size $\lceil \frac{z}{n^{1-\delta}} \rceil$ as follows, for every i for which $e_i = 1$ add the vertex i to the independent set. This would give us a $n^{1-\delta}$-approximation for IS in polynomial time. Relying on the same results as in Theorem 3.4.5, this provides the bound of Houtman and Maks' index. □

3.6 Afriat's index

3.6.1 Introductory observations

As with the previous indices, it is our goal to find the maximum value of e ($e_1 = e_2 = \ldots = e_T = e$), such that a given dataset still passes {W,S,G,H}ARP. However, such a maximum value frequently does not exist. For example, consider the following matrix of the values $p_i q_j$ (for two observations), $i, j = 1, 2$:

$$\begin{pmatrix} 1 & 0.50 \\ 0.60 & 1 \end{pmatrix}.$$

As long as $e \in [0; 0.6[$, all axioms of revealed preference will be satisfied, but for $e \geq 0.6$ a cycle of negative length exists between the two vertices in both $G_e(S)$ and $G_e^H(S)$ and, thus, the axioms are violated. Since there is no maximum feasible value for e, we look for the value e^* that is the supremum of the values of e for which the axioms of revealed preference are satisfied. Varian (1990) describes an approximation algorithm which

approximates e^* to within $(1/2)^t$ by testing t times the axiom under e. In an overview paper, Varian (2006) mentions that it is also easy to calculate e^* exactly and exact values are calculated for AI-GARP in a number of papers, see for instance Choi et al. (2007). However, to the best of our knowledge, no exact polynomial algorithm has been described in the literature. In the next section we provide such a polynomial time exact algorithm for AI-{W,S,G,H}ARP and a separate algorithm for AI-HARP.

3.6.2 Complexity results

Consider the following algorithm for AI-WARP.

Algorithm 2 AI-WARP, (input: $S = \{(p_i, q_i)|i \in N\}$; output e^*)

1: Set $e^* := 1$.
2: **for all** pairs $i, j \in N$ **do**
3: **if** $p_i q_j < e^*$ and $p_j q_i < e^*$ **then**
4: Set $e^* := \max\{p_i q_j, p_j q_i\}$.
5: **end if**
6: **end for**

Theorem 3.6.1. *Algorithm 3 solves* AI-WARP *in* $O(n^2)$ *time.*

Proof. We first argue that Algorithm 2 is correct. Clearly, there is a violation of WARP(e) if and only if there exists a pair of observations $i, j \in N$ $(q_i \neq q_j)$, such that both $p_i q_j \leq e$ and $p_j q_i \leq e$. By construction, the Algorithm ensures that for every pair of observations $k, l \in N$ $(q_k \neq q_l)$ either $p_k q_l \geq e^*$ or $p_l q_k \geq e^*$ or both. It follows that for any $e < e^*$, it is the case that either $p_k q_l > e$ or $p_l q_k > e$ and thus WARP(e) is satisfied. It follows that Algorithm 2 is correct.

The complexity of this Algorithm is straightforward, there exist $O(n^2)$ pair of observations $i, j \in N$. For each of these pairs, $p_i q_j$ and $p_j q_i$ are calculated and compared against e^*, which is done in constant time. Depending on

the outcome of these comparisons, at most one variable is changed, again in constant time. The total number of operations in the Algorithm is thus determined by the number of pairs, $O(n^2)$. □

Next, we consider an algorithm for AI-SARP.

Algorithm 3 AI-SARP, (input: $S = \{(p_i, q_i)|i \in N\}$; output e^*)

1: Initialization: Construct an array A of all values $p_i q_j \leq 1, i \neq j$ and add an element with value equal to 1. Sort all values in ascending order, and let x be the median value in A.
2: Test SARP(x). If SARP(x) is satisfied, remove all values lower than or equal to x from A, otherwise remove all higher values.
3: If more than one element remains in the array, repeat step 2; otherwise x is the remaining value in A. Set $e^* = x$

Theorem 3.6.2. *Algorithm 3 solves* AI-SARP *in* $O(n^2 \log n)$ *time.*

Proof. We first argue that Algorithm 3 is correct. Clearly, if the dataset satisfies SARP(e), then it satisfies SARP(e') for all $e' \leq e$. Moreover, the dataset satisfies SARP(0). Thus, for an increasing e, SARP(e) becomes violated at some value e^*. This can only happen when an arc, completing a cycle, is added to the graph $G_e(S)$, i.e., at some value $p_i q_j$. If there is no value $e < 1$ for which SARP(e) is violated, then $e^* = 1$. It follows that Algorithm 3 is correct.

Next we analyse the complexity of this algorithm. To construct A, $p_i q_j$ must be calculated for all pairs of observations, which takes $O(n^2)$ time. In the worst case, this array is of size $O(n^2)$, so sorting is done in $O(n^2 \log n)$. In the second step of the algorithm, SARP(e) is tested for different values of e. As the array is halved in each iteration, at most $O(\log(n^2))$ such tests are needed and each such test can be done in $O(n^2)$ using, for example, a topological ordering algorithm (Ahuja et al., 1993). This gives a total time complexity for the second step of $O(n^2 \log n)$. The total time complexity

is thus determined by the sorting of the array and the second step and is $O(n^2 \log n)$. □

Algorithm 4 AI-GARP, (input: $S = \{(p_i, q_i) | i \in N\}$; output e^*)

1: Initialization: Construct an array A of all values $p_i q_j \leq 1, i \neq j$ and add an element with value equal to 1. Sort all values in ascending order, and let x be the median value in A.

2: Test GARP(x). If GARP(x) is satisfied, remove all values lower than x from A, otherwise remove all higher values.

3: If more than two element remains in array A, repeat step 2.

4: Let the lowest of the two remaining values be x_1 and the highest x_2. Test GARP(e) for $e = x_1$, and for $e = x_2$. Then

   i. If GARP(x_1) and GARP(x_2) are satisfied, test GARP($x_2+\epsilon$), (where $\epsilon > 0$ is an arbitrarily small number). If GARP($x_2+\epsilon$) is satisfied, then $e^* = 1$ otherwise $e^* = x_2$.

   ii. If both are not satisfied, $e^* = x_1$.

   iii. If GARP(x_1) is satisfied and GARP(x_2) is not, test GARP($x_1 + \epsilon$). If GARP($x_1 + \epsilon$) is satisfied, $e^* = x_2$, otherwise $e^* = x_1$.

In the case of AI-SARP and AI-WARP, an e^* value is computed that corresponds to some arc appearing in the graph, at which point a cycle appears. In GARP(e) however, a cycle does not necessarily indicate a violation since the length of the cycle may be 0. Therefore, we need a subtle change as can be seen in in Algorithm 4.

Theorem 3.6.3. *Algorithm 4 solves* AI-GARP *in* $O(n^2 \log n)$ *time.*

Proof. We first notice that the value e^* can be feasible for GARP(e), if for that value a cycle of length 0 exists in the graph $G(S)$. Therefore, we consider Algorithm 4, which does not discard the highest known feasible value of e.

The time complexity of this algorithm is similar to that for AI-SARP. Throughout most of the algorithm, the only difference is the testing of GARP(e) instead of SARP(e). In the final step, GARP(e) is tested twice, which has no impact on the overall bound. Since testing GARP can be done in $O(n^2)$ time, as shown in Chapter 2, the overall complexity is thus $O(n^2 \log n)$. □

Finally, we provide a polynomial time algorithm for AI-HARP. In this algorithm we need to compute the minimum cycle mean: the cycle with the shortest average arc length.

Algorithm 5 AI-HARP, (input: $S = \{(p_i, q_i)|i \in N\}$; output e^*)

1: Initialization: Construct the graph $G_1^H(S)$.
2: Calculate the minimum cycle mean (MCM), which is the shortest average length of the arcs in any cycle in the graph $G_1^H(S)$.
3: Calculate e^* as follows: $e^* = \exp(MCM)$.

Theorem 3.6.4. *Algorithm 5 solves* AI-HARP *in* $O(n^3)$ *time.*

Proof. We will show that computing the minimum cycle mean (MCM) of $G_1^H(S)$ is sufficient to find e^*. HARP(e) is satisfied if there are no cycles of negative length in $G_e^H(S)$. Thus, if such a cycle exists, we need to remove it by lowering e. A decrease in e will lengthen every arc in the graph by the same amount, as the length of an arc is $\log(p_i q_j) - \log(e)$. It is clear that the if we set the value of e^* so that the cycle with the shortest average arc length has a length of zero, the average arc length of every other cycle will be non-negative and no cycles of negative length will remain. Indeed, by setting $e^* := \exp(\text{MCM})$, the length of each arc becomes $\log(p_i q_j) - \log(\exp(\text{MCM})) = \log(p_i q_j) -$ MCM.

The time complexity of this algorithm is polynomial as there exist algorithms for finding the MCM in $O(nm)$ time (Karp, 1978), with m being the

number of arcs in the graph. In $G_1^H(S)$ there will be n^2 arcs, as there exist arcs from every vertex to every other vertex in the graph. The building of the graph in the initialization step takes $O(n^2)$ time. The overall time bound of the algorithm is thus $O(n^2 \log n + n^3) = O(n^3)$ time. □

3.7 Conclusion

Motivated by the increasing availability of large scale consumption datasets, and by the observed empirical difficulty of computing goodness-of-fit measures, we have investigated the computational complexity of testing the utility maximization hypothesis in revealed preference terms. In particular, we have focused on three goodness-of-fit measures for four different revealed preference axioms (i.e. WARP, SARP, GARP and HARP). We have demonstrated that, for all four axioms, both Varian's and Houtman and Maks' index is inapproximable. Next, we have shown that these conclusions do not apply to Afriat's index, and we have presented exact polynomial algorithms for computing this index (for every revealed preference axiom that we considered).

There are different avenues for further research. Clearly, when insisting on optimal solutions for one of the indices considered here (except Afriat), one needs to accept long running times. It is therefore interesting to develop bounds: not only heuristics should be defined and tested, but also the design of good upper bounds for the indices is an interesting (and largely unexplored) issue. Another direction is to consider datasets or utility maximizations models that are in some way restricted. Essentially this may ensure that the graphs built in our analysis simplify such that computation of the respective indices becomes easier.

Chapter 4

Goodness-of-fit Measures: The Money Pump Index

4.1 Introduction

In the previous chapter, we considered a number of classical goodness-of-fit measures for revealed preference tests. As we noted in Chapter 1, there is continuing interest in these measures. In particular, a number of new goodness-of-fit measures have been proposed in the literature over the past few years. In this chapter, we will look at one of these measures, the *Money Pump Index* (MPI), proposed by Echenique et al. (2011). The MPI is based on the idea that irrational behavior makes consumers vulnerable, as it allows arbitrageurs to "pump money" from them. In particular, arbitrageurs can extract money from irrational consumers by following an opposite purchasing strategy. Echenique et al. (2011) propose to use the amount of money a consumer can lose in this way as a measure of the irrationality of his behaviour.

This chapter is the result of a collaboration with Laurens Cherchye, Bram De Rock and Frits C.R. Spieksma. An article based on parts of this chapter has been published in The Journal of Political Economy (Smeulders et al., 2013). We gratefully acknowledge helpful remarks from Federico Echenique.

This chapter is concerned with the practical computation of the MPI. As we will explain below, if a consumer violates rationality, then typically there will be multiple purchase observations implying such a violation. In principle, we can compute a money pump cost for each violation. This calls for an aggregate MPI that summarizes these money pump costs into a single metric. In their original contribution, Echenique et al. (2011) propose the mean and median money pump cost as such aggregate MPIs. Obviously, these Mean and Median MPIs have an intuitive interpretation in terms of the money lost by the consumer due to irrational behaviour. A first contribution of this note is that we show that computing the Mean and Median MPIs is a #P-COMPLETE problem. As a polynomial time algorithm for a #P-COMPLETE problem would imply P = NP, this result provides a formal statement of the fact that it is computationally challenging to compute these measures in practice, in particular for datasets with large numbers of observations.

Notable examples of such large datasets are household-level "scanner" datasets, which Echenique et al. (2011) also considered in their empirical application. Scanner datasets contain information on household-level purchases collected at checkout scanners in supermarkets. They typically consist of multiple purchase observations for many households. Such large datasets are increasingly available, and Echenique et al. provide a particularly convincing case on the usefulness of their MPI concept in combination with scanner data. At this point, however, it is worth emphasizing that they also extensively discussed the computational complexity of the MPI for their own scanner dataset (see in particular their Remark 1 on p. 1207). To mitigate the computational burden, they therefore suggested as a practical method to compute approximations of the Mean and Median MPIs. Essentially, these approximations focus on violations of revealed preference axioms that involve only a small number of observations (see Section 4.4 for more details).

Because of the computational difficulties associated with the Mean and Median MPIs, our second contribution is that we propose the Maximum

and Minimum MPIs as easy-to-apply alternatives. The Maximum MPI gives the percentage of money lost in the most severe violation of rationality, while the Minimum MPI does the same for the least severe violation. Clearly, these measures preserve the intuition underlying the Mean and Median MPIs. In particular, they figure as natural bounds on the amounts of money that an arbitrageur can extract from irrational consumers.

Importantly, our newly proposed Maximum and Minimum MPIs have clear practical usefulness. We show that the Maximum and Minimum MPIs can be computed efficiently (i.e. in polynomial time), which makes them easily applicable to large (e.g. scanner) datasets. We also indicate how such computation can proceed in practice. Next, we use the dataset of (Echenique et al., 2011) to demonstrate the application of the Maximum and Minimum MPIs. Here, our particular focus is on assessing the performance of these measures relative to the Mean and Median MPIs. In addition, we show that comparing the values of the Maximum and Minimum MPIs can reveal interesting information to the empirical analyst. This makes us believe that our results may contribute to the further dissemination of the intuitive MPI concept in empirical analyses of (ir)rational consumer behaviour.

The rest of this note unfolds as follows. Section 4.2 introduces the MPI concept and the associated notions of Mean and Median MPI. Section 4.3 contains our core results, it introduces Maximum and Minimum MPI and defines the computational complexity of the different MPIs that we consider. Section 4.4 shows the practical usefulness of our results through an application to the scanner dataset of Echenique et al. Section 4.5, finally, concludes.

4.2 Money Pump Index

As in previous chapters, we consider a dataset $S = \{(p_i, q_i)|i \in N\}$, of n observed purchases by a consumer, with prices $p_i \in \mathbb{R}^m_{++}$ and quantities

$q_i \in \mathbb{R}^m_+$ for every observation $i = 1, \ldots, n$. We denote by $q_{i,j}$ ($p_{i,j}$) the quantity (price) of the j-th good in observation t.

As explained by Echenique et al. (2011), if GARP is violated, a money pump cost (MPC) can be calculated for every violation. This MPC is the amount of money an arbitrageur could gain from the consumer by following an appropriate trading strategy. More precisely, suppose that we have two observations i and j for which $p_i q_i \geq p_i q_j$ and $p_j q_j > p_j q_i$. This implies a violation of GARP that involves the observations i and j. Then, the arbitrageur can make money by buying bundle q_i at prices p_j and reselling it at p_i, and by buying q_j at prices p_i and reselling it at p_j. The total profit following from these transactions gives the corresponding MPC, which equals

$$p_i(q_i - q_j) + p_j(q_j - q_i). \tag{4.1}$$

Generalizing this argument, we can compute the MPC associated with a GARP violation involving a sequence of observations $v_1, v_2, \ldots, v_k$ as follows

$$MPC = \sum_{j=1}^{k} p_{v_j}(q_{v_j} - q_{v_{j+1}}), \tag{4.2}$$

with $q_{v_{k+1}} = q_{v_1}$.

To be able to make meaningful comparisons between GARP violations involving different sequences, the Money Pump Index (MPI) of a violation is calculated by dividing the associated MPC by the total budget of the observations that are involved in the violation. That is

$$MPI = \frac{\sum_{j=1}^{k} p_{v_j}(q_{v_j} - q_{v_{j+1}})}{\sum_{j=1}^{k} p_{v_j} q_{v_j}}, \tag{4.3}$$

with again $q_{v_{k+1}} = q_{v_1}$.

If a dataset for a given consumer violates GARP, then there are typically several sequences of observations that are involved in a violation. Therefore, Echenique et al. (2011) introduce the Mean and Median MPI of consumers as measures of consumer irrationality. More precisely, each violation gives rise to an MPI value (as defined in (4.3)). The Mean MPI is then

defined as the mean of these MPI values, while the Median MPI equals the median of these values. These measures indeed have an intuitive meaning as quantifying the severity of consumer irrationality.

4.2.1 Graph Representation

As in previous chapters, we will represent the graph associated with a dataset S by $G(S) = (V, A)$. There exists a vertex $i \in V$ for every observation $i = 1, \ldots, n$ in the dataset S. An arc $(i, j) \in A$ exists if and only if $p_i q_i \geq p_i q_j$. The length of an arc $\ell(i, j)$, is given by $p_i q_i - p_i q_j$. Beside a length, arcs are now also given a weight $w(i, j) = p_i q_i$, which is equal to the expenditures in observation i. If there is a sequence of observations $v_1, v_2, \ldots, v_k$, such that there is a cycle $(v_1, v_2), (v_2, v_3), \ldots, (v_k, v_1)$, we call this a cycle in the graph, and denote it by C. By construction, the following holds.

$$MPI = \frac{\sum_{j=1}^{k} p_{v_j}(q_{v_j} - q_{v_{j+1}})}{\sum_{j=1}^{k} p_{v_j} q_{v_j}} = \frac{\sum_{j:v_j \in C} \ell(v_j, v_{j+1})}{\sum_{j:v_j \in C} w(v_j, v_{j+1})} \tag{4.4}$$

In what follows, we will denote the MPI of a given violation or cycle by $MPI(C)$.

4.3 Complexity results

In their original contribution, Echenique et al. already argued that computing the Mean and Median MPI is a challenging task and, therefore, they propose to approximate these MPIs in practical applications (see also Section 4.4). In what follows, we will formally state that computing the Mean and Median MPI is indeed a #P-COMPLETE problem.

To demonstrate that the Mean and Median MPI are computationally hard, we derive a reduction from the #CYCLE problem. We first give the definition of this problem.

Problem 4.3.1. *#Cycle*
Instance: A graph $G = (V, A)$.
Question: How many cycles are contained in the graph G ?

Valiant (1979) shows that the #CYCLE problem is #P-COMPLETE. Thus, our reduction implies that computing the Mean and Median MPIs is also #P-COMPLETE, as such, a polynomial time algorithm for these problems implies P = NP.

Theorem 4.3.1. *Calculating the Mean MPI is a* #P-COMPLETE *problem.*

Proof. Consider an instance of the problem #CYCLE, that is we have a directed graph $G = (V, A), |V| = n$ with the question: how many directed cycles exist in G ? We will answer the question by computing the Mean MPI of two specially constructed sets of consumer data S_1 and S_2. Both datasets consist of $n+2$ observations and $m+2$ goods. In fact, observations $1, 2, \ldots, n$ are identical for S_1 and S_2 and can be described as follows.

For each vertex $i \in V$, we construct a price vector p_i with $p_{i,j} = \epsilon$ for $i \neq j$ ($\epsilon < \frac{1}{2n}$) and $p_{i,i} = 1$. For every vertex i we create a quantity vector q_i with $q_{i,i} = 1$, $q_{i,j} = 0$ if there is an arc from j to i in G (for $i \neq j$), and $q_{i,j} = 2$ if there is no arc (again for $i \neq j$). Observe that an arc in G corresponds to an arc in the graph representation of the dataset consisting of these n observations, and vice versa. We will denote this part of the dataset by $\bar{S}$. We now finish the description of S_1, by specifying observations $n+1$ and $n+2$ as follows. Let $p_{n+1} = (2, 1, 1, \ldots, 1)$, $p_{n+2} = (1, 2, 1, \ldots, 1)$, $q_{n+1} = (3, 2, 2, \ldots, 2)$ and $q_{n+2} = (2, 3, 2, \ldots, 2)$. Notice that no observation $\{1, 2, \ldots, n\}$ is preferred over observation $n + 1$ and $n + 2$. Further notice that observation $n + 1$ is preferred over $n + 2$, and vice versa. Hence, the number of cycles in $G(S_1)$ is 1+ the number of cycles in G, or 1+ #CYCLE for short. In particular, we can easily verify that the MPI (see (4.3)) of the additional cycle equals $\frac{1}{2n+4}$. Let us write $MPI(C)$ for the value of the MPI corresponding to a cycle C in the graph representation of the dataset S. Then, the mean MPI of dataset S can be written as:

$$MPI = \frac{\sum_{C \in G(S_1)} MPI(C)}{\#Cycle + 1} = MPI = \frac{\sum_{C \in G(\bar{S})} MPI(C) + \frac{2}{2n+4}}{\#Cycle + 1} \quad (4.5)$$

Now, we finish the description of dataset S_2 by specifying observations $n+1$ and $n+2$ as follows: $p_{n+1} = (2, 1, 1, \ldots, 1)$, $p_{n+2} = (1, 2, 1, \ldots, 1)$, $q_{n+1} = (4, 2, 2, \ldots, 2)$ and $q_{n+2} = (2, 4, 2, \ldots, 2)$. As in dataset S_1 there is one additional cycle between nodes $n+1$ and $n+2$, which has MPI equal to $\frac{2}{2n+6}$. Thus, the mean MPI of dataset S_2 can be written as.

$$MPI = \frac{\sum_{C \in G(S_1)} MPI(C)}{\#Cycle + 1} = MPI = \frac{\sum_{C \in G(\bar{S})} MPI(C) + \frac{2}{2n+6}}{\#Cycle + 1} \quad (4.6)$$

Now suppose we have a polynomial time algorithm for finding the mean MPI of a dataset, then we can find the mean MPI for S_1 and S_2, compute the difference and with the knowledge that this difference is $\frac{\frac{2}{2n+6} - \frac{1}{2n+4}}{\#\text{Cycle}+1}$ find #Cycle. This implies that we would have a polynomial time algorithm for solving #Cycle, which in turn implies the Mean MPI problem is #p-complete. □

Theorem 4.3.2. *Calculating the Median MPI is a* #p-complete *problem.*

Proof. Consider an instance of the problem #Cycle. We will solve this problem by computing the median MPI of a polynomial number ($n \log(n)$) of specially constructed sets of consumer data. First, number the vertices of G from 1 to n. We will then construct a dataset S^- with n observations and n goods as follows. For every vertex $i \leq n$ we construct a price vector p_i with $p_{i,j} = \epsilon$ for $j \neq i$ ($\epsilon < \frac{1}{2n^2}$), $p_{i,i} = 1$. For every vertex i we create a quantity vector q_i with $q_{i,i} = 1$, $q_{i,j} = 0$ if there is an arc from j to i in G (for $i \neq j$), and $q_i^j = 2$ if there is no arc (again for $i \neq j$). It can be easily checked that the graph representation of S^- has the same set of arcs as the original graph G. It follows that both have the same number

of cycles. Given the construction, we can compute upper bounds on the budgets and lower bounds on the arc lengths. The budget is maximized if, for observation i, $q_{i,i} = 1$ and $q_{i,j} = 2, (i \neq j)$; the budget then becomes $1 + 2(n-1)\epsilon < 1 + \frac{2(n-1)}{2n^2}$. A lower bound on the arc lengths is given by the combination of a lower bound on $p_i q_i$, which is easily seen to be 1, and an upper bound on $p_i q_j$. This upper bound on $p_i q_j$ is reached if $q_{j,j} = 1$, $q_{j,k} = 2, (i \neq k \neq j)$ and since the arc (i,j) exists, $q_{j,i} = 0$. The upper bound on $p_i q_j$ then equals $2(n-2)\epsilon + \epsilon$, which, in turn, is bounded from above by $\frac{1}{n} - \frac{1}{n^2}$. A lower bound on the arc lengths is then $1 - (\frac{1}{n} + \frac{1}{n^2})$. Given these bounds, for any given arc (i,j), it is the case that $\frac{\ell(i,j)}{w(i,j)} = \frac{n^2-(n-1)}{n^2+n-1} > 0.5$ and thus $\frac{n^2-n+1}{n^2+n-1} > 0.5$ is also a lower bound on the MPI of any given cycle. Finally, this is also a lower bound on the minimum MPI of the dataset S^-.

We now add 2 more observations and 2 goods to S^-, creating S. For every observation $i < n+1$, $p_{i,n+1} = p_{i,n+2} = 2$ and $q_{i,n+1} = q_{i,n+2} = 0$. Set $p_{n+1} = (\epsilon, \ldots, \epsilon, 1, 0.5)$, $p_{n+1} = (\epsilon, \ldots, \epsilon, 0.5, 1)$, $q_{n+1} = (0, \ldots, 0, 1, 0)$ and $q_{n+1} = (0, \ldots, 0, 0, 1)$. It is clear that $n+1$ and $n+2$ are preferred over every other observation and that no observation $i < n$ is preferred over either $n+1$ or $n+2$. In this way, one more violation is added, with an MPI of 0.5. It follows that the minimum MPI of dataset S has a value of 0.5 and that there is one unique violation that has this MPI.

Now, consider that we add additional goods and observations, denoted by D, to the dataset S, creating $S \cup D = S^+$. For these new goods and observations, the prices and quantities are so that all existing violations remain, and have the same MPIs, while a known number of new violations are created, and the MPIs of these new violations are smaller than the minimum MPI of violations in S. It is clear that if the Median MPI of S^+ is equal to the minimum MPI of S, then the number of new violations created in S^+ is equal to the number of violations in D and thus one more than the number of cycles in G. We will now show that we can efficiently add new goods and observations to S to create a known number of extra violations

in S^+, and that creating a polynomial number of datasets is sufficient for finding a S^+ for which the median MPI is equal to the minimum MPI of S.

First, we notice that G has less than $O(n \times n!) < O(n^{n+1})$ cycles. A binary search over this number can be done in $O(\log(n^{n+1})) = O((n+1)\log(n))$ time. At each step in this binary search, we add a component D to S so that $S \cup D = S^+$. This component is created as follows. Let $f(k)$ be the number of cycles in a fully connected digraph with k nodes. Now assume x arcs must be added to S to form S^+, then find $\max_k(f(k) < x)$ and add $\left\lfloor \frac{x}{f(k)} \right\rfloor$ subcomponents of k observations to D. The prices and quantities of these observations are set such that all observations within one subcomponent are preferred over all other observations in that subcomponent, and so that the MPIs of these violations are smaller than the minimum MPI of S and so that no cycles that include observations of multiple subcomponent exist. This is done as follows; we use a number of goods equal to the number of observations, and rank all the subcomponents. For every observation i added, set $q_{i,i} = 1$, $q_{i,j} = 0$ otherwise. Set $p_{i,i} = 1$, $p_{i,j} = 0.75$ if j is associated with another observation in the same subcomponent, and if j is associated with an observation in a higher-ranked subcomponent $p_{i,j} = 2$. Finally, if j is associated with an observation in a lower-ranked subcomponent or D, set $p_{i,j} = \epsilon$. It is easy to see that for a given x we can efficiently find the groups to be added and, as $(2k+1) \times f(k) > f(k+1)$, the number of subcomponents is polynomial.

In conclusion, the #Cycle problem for a graph G can be solved by calculating the Median MPI of at most $O((n+1)\log(n))$ graphs, which can be constructed in polynomial time and have a size that is polynomial in the size of the graph G. As such, a polynomial time algorithm for the Median MPI would mean a polynomial time algorithm for #Cycle, which in turn implies the Median MPI problem is #P-complete. □

In these proofs, the number of goods, m, is not bounded by a constant. Hence, it remains an open question whether or not polynomial-time algo-

rithms exist when the number of goods is fixed. Recent results by Deb and Pai (2013) show that, given a large number of observations compared to the number of goods, some structure will appear in the preference relations. It is possible that this structure can be exploited to find polynomial time algorithms in these cases.

We suggest using the Maximum and Minimum MPIs as easy-to-apply alternative measures of irrationality. These measures are calculated as, respectively, the maximum and minimum MPI values defined over all violations. Interestingly, we can prove that these Maximum and Minimum MPIs can be computed in polynomial time, which makes them particularly attractive from an empirical point of view.

Theorem 4.3.3. *The time required to compute the Maximum MPI and the Minimum MPI is polynomial in the number of observations.*

Proof. The proof of this theorem follows directly from the graph representation. Given the length and weights assigned to arcs, the problem of finding a Maximum MPI is equivalent to the known Minimum Cycle Ratio problem. (See Ahuja et al. (1993), notice that every arc has negative length, and the worst violation, the maximum MPI, thus corresponds to the cycle with the minimum cycle ratio.) Since constructing the graph is possible in $O(n^2)$ time, and Megiddo (1979) showed that computing the Minimum Ratio Cycle has a time complexity of $O(n^3 \log n)$, the theorem follows. The Minimum MPI can likewise be found in polynomial time, by choosing the length of the arcs to be $p_i q_j - p_i q_i$. □

4.4 Empirical application: deterministic test results

We next compute the newly proposed Maximum and Minimum MPIs for the dataset reported in Echenique et al. (2011). This dataset contains 494 households (i.e. 494 consumers), with 26 purchase observations per

household. Out of these 494 households, there are 396 that violate GARP. The numbers reported in Table 4.1 pertain to this subset of households.

To compute our results for the Maximum and Minimum MPIs, we implemented an algorithm described in Ahuja et al. (1993) for solving the Minimum Cycle Ratio problem. This algorithm is very quick in practice: we needed only a few seconds to compute the results for all 494 households.

Table 1 presents summary statistics on the different MPIs under consideration. Let us first consider our findings for the maximum and minimum MPIs. As indicated in the Introduction, we believe these results reveal interesting information, as they give the maximum and minimum amounts of money that an arbitrageur can extract from irrational consumers. We find that the average Maximum MPI equals 9.35%, while the average Minimum MPI amounts to 3.41%. However, the corresponding standard deviations also reveal that these numbers hide quite some variation across households. Next, we observe that the range between the Maximum and Minimum MPIs is on average 5.95%, and that this range also varies quite substantially across households. In this respect, however, it is also worth noting that the range turns out to be zero for no less than 74 households, i.e. for about one-fifth of the 396 households exhibiting violations of GARP, we obtain that the Maximum MPI exactly equals the Minimum MPI.

As a final base of comparison, we compare our results to the ones reported by Echenique et al. (2011). As indicated above, these authors recognized the complex nature of computing the Mean and Median MPIs and therefore resorted to computing approximations of these MPIs in their empirical application. In particular, they approximated the Mean and Median MPIs by focusing on short violations only, i.e. violations consisting of at most four observations. Table 1 reports the associated descriptive statistics. When comparing Echenique et al.'s results for the Mean and Median MPIs to the ones for our Maximum and Minimum MPIs, we conclude that, in many cases, these last two "extreme" MPIs spread symmetrically around the first two "central" MPIs. This suggests that the average of the Maximum and

Minimum MPIs may actually provide relevant information. In particular, these numbers can be used to obtain a good estimate of the Mean and Median MPIs.

4.5 Conclusion

We have shown that the Mean and Median MPIs originally proposed by Echenique et al. (2011) are generally difficult to compute (#P-COMPLETE), which makes them impractical in the case of large datasets. As alternatives, we therefore proposed the Maximum and Minimum MPIs. These MPIs can be computed efficiently (i.e. in polynomial time) and preserve the attractive interpretation of the Mean and Medium MPI. We also demonstrated the practical usefulness of these Maximum and Minimum MPIs through an application to the scanner dataset which Echenique et al. also studied. We hope that our results will contribute to the further dissemination of the intuitive MPI concept in empirical analyses of (ir)rational consumer behaviour.

Table 4.1: *Descriptive statistics for alternative MPIs*

	Minimum MPI	Maximum MPI	Range	Approximated Mean MPI	Approximated Median MPI
Average	0.0341	0.0936	0.0595	0.0610	0.0591
Standard Deviation	0.0287	0.0616	0.0603	0.0359	0.0369
Number of zeros	0	0	74	0	0
Minimum	0.0002	0.0048	0.0000	0.0048	0.0048
First quartile	0.0154	0.0489	0.0115	0.0355	0.0322
Median	0.0268	0.0797	0.0443	0.0543	0.0497
Third quartile	0.0429	0.1274	0.0867	0.0809	0.0791
Maximum	0.2782	0.4010	0.3350	0.2782	0.2782

Chapter 5

Transitive Preferences in Multi-Member Households

5.1 Introduction

In the previous chapters, we studied rationalizability by unitary decision makers. In this chapter and the next, we change focus and study consumption decisions (including labour supply decisions) of multi-member households. By now, it is well established that the collective model of Chiappori (1988) is both conceptually and empirically attractive for analyzing consumption behavior (see, for example, Vermeulen (2002) for an overview of the relevant literature). This collective model assumes that the different household members are endowed with individual preferences defined over privately and publicly consumed goods (inside the household). These members enter into a decision process of which the outcome is assumed to obtain a Pareto optimal allocation (of the aggregate household budget). In what follows, we say that multi-person household behavior is *collectively rational* if it is consistent with the collective model, see Section 5.2.

This chapter is the result of a collaboration with Laurens Cherchye, Bram De Rock, Fabrice Talla Nobibon and Frits C.R. Spieksma. An article based on this chapter is accepted for publication in Economic Theory Bulletin (Smeulders et al., 2014a).

In the tradition of Afriat (1967b) and Varian (1982), we are interested in the revealed preference characterization of collective models. Such a revealed preference characterization does not rely on any functional specification regarding the household consumption process, and starts directly from the observed finite set of prices and quantities. Varian (1982) introduced the revealed preference axioms that summarize the empirical implications of theoretical consumption models for single-member households, while Cherchye et al. (2011) provided a revealed preference characterization of collective models for multi-member households. See also Peters and Wakker (1994), Varian (2006), Cherchye et al. (2007), Cherchye et al. (2010) and Cherchye et al. (2013) for more discussion.

Our following analysis focuses on two popular revealed preference axioms: the *Weak Axiom of Revealed Preferences* (WARP) introduced by Samuelson (1938), and the *Strong Axiom of Revealed Preferences* (SARP) introduced by Houthakker (1950). For single-person households, these axioms summarize the testable implications of rational (i.e. utility maximizing) consumption behavior. Essentially, SARP extends WARP by requiring preferences to be transitive. In this respect, a classical result due to Rose (1958) shows that WARP and SARP are empirically equivalent in a setting with 2 goods. In other words, transitivity has no empirical bite if the analysis includes only 2 goods. Because WARP is generally easier to test than SARP, this result can considerably facilitate the empirical analysis.

In what follows, we investigate the possibility to extend the result of Rose (1958) towards multi-member households. Therefore, we define the concepts of L-WARP and L-SARP, which capture the testable implications of collectively rational (i.e. Pareto efficient) consumption behavior in the case of L household members. In a first instance, we assume a general setting in which we only observe the aggregate household consumption, i.e. no information is available on the intrahousehold allocation of the private goods. Here, we obtain two main results. First, we show that L-WARP and L-SARP are empirically vacuous (i.e. non-rejectable) if there are no more than L goods. Next, and more importantly, we show that WARP and SARP

are not equivalent if $L > 1$ and there are at least $L+1$ goods. Thus, Rose's conclusion does not generalize to L-member households (even with as few as four observations). Transitivity of individual preferences is a testable requirement even if there are only $L + 1$ goods for L household members.

Finally, we also study a more restricted setting where we exclusively assign a single good to each different household member, i.e. L (out of $L+1$) goods are exclusive. Thus, in contrast to the general setting, this restricted setting assumes that we observe the intrahousehold allocation of L goods. We call this a "labour supply" setting as it formally coincides with Chiappori's original labour supply model, in which each household member exclusively consumes his/her (observed) leisure while the remaining consumption is captured by a Hicksian aggregate (that is to be shared among the household members) (Chiappori, 1988). Interestingly, we can show that L-WARP and L-SARP do become equivalent under these empirical conditions (with $L + 1$ goods). Thus, the empirical analysis need not explicitly account for transitivity, which can substantially alleviate the computational burden in practical applications.

The rest of the chapter unfolds as follows. In Section 5.2 we introduce the collective model and the corresponding revealed preference axioms. In Section 5.3 we investigate the equivalence between L-WARP and L-SARP for both the general setting and the restricted labour supply setting. Section 5.4 concludes.

5.2 Notation and Definitions

We consider an L-member household that consumes m_1 private goods and m_2 public goods (with $L, m_1, m_2 \in \mathbb{N}_0$). The vector $\mathbf{q} \in \mathbb{R}^{m_1}_+$ represents the quantities that are privately consumed by the household, i.e., these goods are assigned to members of the household, and only the amount assigned to an individual member has an effect on the member's utility. $\mathbf{p} \in \mathbb{R}^{m_1}_+$ stands for the corresponding price vector. Similarly, the vector

$\mathbf{Q} \in \mathbb{R}_+^{m_2}$ represents the publicly consumed quantities, these goods are not allocated to individual members, rather they have an effect on the utility of every household member. $\mathbf{P} \in \mathbb{R}_+^{m_2}$ gives the price vector for the public goods. Next, the vector $\mathbf{q}^\ell \in \mathbb{R}_+^{m_1}$ contains the privately consumed quantities for each individual member ℓ, with $\sum_{\ell=1}^{L} \mathbf{q}^\ell = \mathbf{q}$. The collective model of household consumption explicitly recognizes the individual preferences of the household members. These preferences may depend on the private quantities, the public quantities, or both. Throughout, we assume that preferences of member ℓ can be represented by a well-behaved (i.e. continuous, positive monotonic and concave) utility function $U^\ell(\mathbf{q}^\ell, \mathbf{Q})$, $\ell = 1, \ldots, L$.

Our analysis starts from the dataset $S = \{(\mathbf{p}_t, \mathbf{P}_t; \mathbf{q}_t, \mathbf{Q}_t), t = 1, ..., n\}$, which contains n household choices that are characterized by prices $\mathbf{p}_t$,$\mathbf{P}_t$ and quantities $\mathbf{q}_t$,$\mathbf{Q}_t$. In our general setting, we do not know which quantities are privately consumed by which member, i.e. $\mathbf{q}_t^\ell$ is unobserved. Therefore, we need to introduce (unobserved) feasible personalized quantities that comply with the (observed) aggregate quantities $\mathbf{q}_t$. That is, we consider all possible decompositions $\mathbf{q}_t^\ell \in \mathbb{R}_+^N$ that satisfy $\sum_{\ell=1}^{L} \mathbf{q}_t^\ell = \mathbf{q}_t$. In what follows, our main focus will be on this general setting. However, as indicated in the previous section, we will also consider a restricted (labour supply) setting that is characterized by exclusive goods. An exclusive good is a private good that is exclusively consumed by a given member. Evidently, this setting implies extra information on $\mathbf{q}_t^\ell$.

A collective rationalization of a set of observations S requires the existence of member-specific utility functions for which each observed quantity bundle can be characterized as Pareto efficient. The following definition provides a formal statement.

Definition 5.2.1. *Let $S = \{(\mathbf{p}_t, \mathbf{P}_t; \mathbf{q}_t, \mathbf{Q}_t), t = 1, ..., n\}$ be a set of observations. Then, the utility functions $U^1, \ldots, U^L$ provide a collective rationalization of S if, for each observation t, there exist feasible personalized quantities $\mathbf{q}_t^\ell$ such that:*

1. $\sum_{\ell=1}^{L} \mathbf{q}^{\ell}{}_t = \mathbf{q}_t$.

2. *For all possible quantities* $\mathbf{z}^{\ell}, \mathbf{Z}$ *with* $\sum_{\ell=1}^{L} p_t q_t^{\ell} + P_t Q_t \geq \sum_{\ell=1}^{L} p_t z^{\ell} + P_t Z$, *if there exists a member* ℓ *for whom* $U^{\ell}(\mathbf{z}^{\ell}, \mathbf{Z}) > U^{\ell}(\mathbf{q}_t^{\ell}, \mathbf{Q}_t)$ *then there is some member* m *for which* $U^{m}(\mathbf{z}^{m}, \mathbf{Z}) < U^{m}(\mathbf{q}_t^{m}, \mathbf{Q}_t)$.

Our revealed preference characterizations of collectively rational behavior make use of the concepts L-WARP and L-SARP, which provide multi-member extensions of the WARP and SARP concepts that apply to single-member households. To formally define L-WARP and L-SARP, we need the concept of feasible personalized prices. These are prices $\mathbf{P}^{\ell} \in \mathbb{R}_{+}^{m_2}$ such that $\sum_{\ell=1}^{L} \mathbf{P}^{\ell} = \mathbf{P}$. Intuitively, these personalized prices capture the fractions of the household prices for the public goods that are borne by the individual members ℓ. Given the Pareto efficiency assumption that underlies the collective consumption model, these prices can also be interpreted as Lindahl prices. We refer to Cherchye et al. (2011) for a detailed discussion.

Assume that we observe a dataset $S = \{(\mathbf{p}_t, \mathbf{P}_t; \mathbf{q}_t, \mathbf{Q}_t), t = 1, ..., n\}$, and consider a given specification of feasible personalized quantities $\mathbf{q}_t^{\ell}$ and prices $\mathbf{P}_t^{\ell}$. Then, for household member ℓ, we say that the consumption allocation s is directly revealed preferred over another allocation t (denoted $sR_0^{\ell}t$) if $\mathbf{p}_s\mathbf{q}_s^{\ell} + \mathbf{P}_s^{\ell}\mathbf{Q}_s \geq \mathbf{p}_s\mathbf{q}_t^{\ell} + \mathbf{P}_s^{\ell}\mathbf{Q}_t$. The transitive closure of this relation is denoted by R^{ℓ}. Essentially, the relation R^{ℓ} extends R_0^{ℓ} by exploiting transitivity of individual preferences. We can now define our concepts L-WARP and L-SARP.

Definition 5.2.2. *Let* $S = \{(\mathbf{p}_t, \mathbf{P}_t; \mathbf{q}_t, \mathbf{Q}_t), t = 1, ..., n\}$ *be a set of observations.*

1. S *satisfies* L-WARP *if and only if there exist, for all* $\ell = 1, \ldots, L$ *and for all pairs of observations* $s, t = 1, ..., n$, *feasible personalized quantities* $\mathbf{q}_t^{\ell}$ *and feasible prices* $\mathbf{P}_t^{\ell}$, *such that* $tR_0^{\ell}s$ *implies either* $\mathbf{p}_s\mathbf{q}_s^{\ell} + \mathbf{P}_s^{\ell}\mathbf{Q}_s < \mathbf{p}_s\mathbf{q}_t^{\ell} + \mathbf{P}_s^{\ell}\mathbf{Q}_t$ *or* $(\mathbf{q}_t^{\ell} = \mathbf{q}_s^{\ell}$ *and* $\mathbf{Q}_t = \mathbf{Q}_s)$.

2. *S satisfies L-SARP if and only if there exist, for all $\ell = 1, \ldots, L$ and for all pairs of observations $s, t = 1, ..., n$, feasible personalized quantities $\mathbf{q}_t^\ell$ and feasible prices $\mathbf{P}_t^\ell$, such that $tR^\ell s$ implies either $\mathbf{p}_s\mathbf{q}_s^\ell + \mathbf{P}_s^\ell\mathbf{Q}_s < \mathbf{p}_s\mathbf{q}_t^\ell + \mathbf{P}_s^\ell\mathbf{Q}_t$ or $(\mathbf{q}_t^\ell = \mathbf{q}_s^\ell$ and $\mathbf{Q}_t = \mathbf{Q}_s)$.*

If $L = 1$, Definition 5.2.2 reduces to the standard definition of WARP and SARP in Varian (1982). When $L \geq 2$, then this definition states that, for the given specification of feasible personalized quantities and prices, S satisfies L-WARP if and only if, for each member ℓ, the feasible personalized prices and quantities satisfy WARP. A directly similar interpretation applies to L-SARP, except that this concept also accounts for (indirect) revealed preference relations that are induced by transitivity.

As discussed extensively in Varian (1982) and Varian (2006), WARP defines a necessary condition for the existence of a well-behaved utility function for single-member households. In general, however, WARP is not sufficient because it does not impose transitivity. By contrast, SARP defines a necessary as well as sufficient condition. These insights extend to the multi-member setting that we consider here. In particular, the results of Cherchye et al. (2011) are easily adapted to show that there exist utility functions that provide a collective rationalization of S if and only if at least one specification of feasible personalized quantities and feasible personalized prices satisfies L-SARP. Again, L-WARP provides a corresponding necessary condition by not requiring transitivity of the individual preferences.

5.3 Is L-WARP equivalent to L-SARP?

In this section, we compare the testable implications of L-WARP and L-SARP. We start by showing that L-WARP and L-SARP are empirically vacuous conditions if there are no more than L goods in subsection 5.3.1. Subsequently, we demonstrate that, for $L+1$ goods (or more), the equivalence between L-WARP and L-SARP breaks down for the general setting (without exclusive goods) in subsection 5.3.2. Finally, in subsection 5.3.3 we also

show that L-WARP and L-SARP do become equivalent for the restricted labour supply setting (where L out of the $L+1$ goods are exclusive).

5.3.1 At most L goods

Before presenting our results, let us first recall that for single-member households WARP and SARP are idle conditions if there is only a single good. Indeed, in that case $t\ R_0\ s$ is equivalent to $q_t \geq q_s$ (with $q_t, q_s \in \mathbb{R}_+$ and $s, t = 1, ..., n$), which implies that we can never reject either WARP or SARP. This non-testability result can be extended to L-WARP and L-SARP, as follows.

Proposition 5.3.1. *Let $S = \{(\mathbf{p}_t, \mathbf{P}_t; \mathbf{q}_t, \mathbf{Q}_t), t = 1, ..., n\}$ be a set of observations. Then L-WARP and L-SARP are vacuous conditions as soon as $L \geq m_1 + m_2$.*

Proof. To show this result, we consider the following specification of feasible personalized quantities and prices. For all $t = 1, \ldots, n, i = 1, \ldots, m_1, j = 1, \ldots, L$:

$$\mathbf{P}^i_{t,j} = 0, \mathbf{q}^i_{t,i} = \mathbf{q}_{t,i} \text{ and } \mathbf{q}^i_{t,j} = 0 \text{ if } i \neq j;$$

for all $t = 1, \ldots, n, i = m_1 + 1, \ldots, m_1 + m_2, j = 1, \ldots, L$:

$$\mathbf{q}^i_{t,j} = 0, \mathbf{P}^i_{t,i} = \mathbf{P}_{t,i} \text{ and } \mathbf{P}^i_{t,j} = 0 \text{ if } i \neq j;$$

and, if $L > m_1 + m_2$, for all $t = 1, \ldots, n, i = m_1 + m_2 + 1, \ldots, L, j = 1, \ldots, L$:

$$\mathbf{P}^i_{t,j} = 0 \text{ and } \mathbf{q}^i_{t,j} = 0.$$

For this specification, one can easily verify that for each $\ell = 1, \ldots L$, there is at most one good in the WARP (respectively SARP) test. This shows that L-WARP (respectively L-SARP) is a vacuous condition in this setting.

□

Proposition 5.3.1 implies that we can only meaningfully check consistency with L-WARP and L-SARP if the number of goods $m_1 + m_2$ is strictly larger

than L. Next, it is possible to generalize Example 5.3.1 of Cherchye et al. (2007) to show that both L-WARP and L-SARP can be rejected as soon as there are $L+1$ goods. Given this, a natural next question is whether L-WARP and L-SARP are equivalent for $L+1$ goods. This would generalize the result of Rose (1958) (which shows equivalence for $L = 1$) towards $L \geq 2$.

5.3.2 L-WARP and L-SARP are not equivalent.

For the general setting, the answer to this equivalence question is negative. We can show this for $L = 2$ by means of the dataset in Example 1.

Example 5.3.1. *Let us consider a dataset consisting of the four observations presented in Table 5.1.*

$\mathfrak{q}_1 = (5, 0, 0)$	$\mathfrak{p}_1 = (7.5, 0.5, 0.5)$
$\mathfrak{q}_2 = (0, 5, 0)$	$\mathfrak{p}_2 = (1, 2, 0.9)$
$\mathfrak{q}_3 = (0, 0, 5)$	$\mathfrak{p}_3 = (0.2, 2.02, 2)$
$\mathfrak{q}_4 = (4, 3, 1)$	$\mathfrak{p}_4 = (1, 1, 5)$

Table 5.1: *Example dataset.*

We use this dataset to consider two cases. In the "private case" there is only private consumption, i.e. $\mathbf{q}_t = \mathfrak{q}_t$ *and* $\mathbf{Q}_t = 0$ *for* $t = 1, 2, 3, 4$. *By contrast, in the "public case" there is only public consumption, i.e.* $\mathbf{Q}_t = \mathfrak{q}_t$ *and* $\mathbf{q}_t = 0$ *for* $t = 1, 2, 3, 4$.

For this specific dataset we obtain the following result.

Lemma 5.3.1.

1. *For both the private and the public case, the dataset in Example 5.3.1 does not satisfy* 2-SARP.
2. *For both the private and the public case, the dataset in Example 5.3.1 does satisfy* 2-WARP.

Proof. To prove part 1 of the lemma, we make us of the scalar products in Table 5.2.

$\mathfrak{p}_1\mathfrak{q}_1 = 37.5$	$\mathfrak{p}_2\mathfrak{q}_2 = 10$ $\mathfrak{p}_2\mathfrak{q}_4 = 10.9$	$\mathfrak{p}_3\mathfrak{q}_3 = 10$ $\mathfrak{p}_3\mathfrak{q}_2 = 10.1$	$\mathfrak{p}_4\mathfrak{q}_4 = 11$ $\mathfrak{p}_4\mathfrak{q}_3 = 25$
$\mathfrak{p}_1(\mathfrak{q}_2 + \mathfrak{q}_3 + \mathfrak{q}_4) = 37$	$\mathfrak{p}_2(\mathfrak{q}_1 + \mathfrak{q}_3) = 9.5$	$\mathfrak{p}_3(\mathfrak{q}_1 + \mathfrak{q}_4) = 9.86$	$\mathfrak{p}_4(\mathfrak{q}_1 + \mathfrak{q}_2) = 10$

Table 5.2: *Some relevant scalar products.*

The numbers of Table 5.2 show that the following inequalities hold: $\mathfrak{p}_1\mathfrak{q}_1 \geq \mathfrak{p}_1(\mathfrak{q}_2+\mathfrak{q}_3+\mathfrak{q}_4), \mathfrak{p}_2\mathfrak{q}_2 \geq \mathfrak{p}_2(\mathfrak{q}_1+\mathfrak{q}_3), \mathfrak{p}_3\mathfrak{q}_3 \geq \mathfrak{p}_3(\mathfrak{q}_1+\mathfrak{q}_4)$, and $\mathfrak{p}_4\mathfrak{q}_4 \geq \mathfrak{p}_4(\mathfrak{q}_1+\mathfrak{q}_2)$. Assume that all consumption is private and consider any specification of feasible personalized quantities $\mathbf{q}_\ell^1$ and $\mathbf{q}_\ell^2$. Then, $\mathfrak{p}_1\mathfrak{q}_1 \geq \mathfrak{p}_1(\mathfrak{q}_2 + \mathfrak{q}_3 + \mathfrak{q}_4)$ implies that there always exists at least one ℓ for which $\mathbf{p}_1\mathbf{q}_1^\ell \geq \mathbf{p}_1(\mathbf{q}_1^\ell + \mathbf{q}_2^\ell + \mathbf{q}_3^\ell)$. Indeed, assume this is not the case, i.e. $\mathbf{p}_1\mathbf{q}_1^\ell < \mathbf{p}_1(\mathbf{q}_1^\ell + \mathbf{q}_2^\ell + \mathbf{q}_3^\ell)$ for both $\ell = 1$ and $\ell = 2$. Adding up these last two inequalities then gives a contradiction: $\mathfrak{p}_1\mathfrak{q}_1 = \mathbf{p}_1\mathbf{q}_1^1 + \mathbf{p}_1\mathbf{q}_1^2 < \mathbf{p}_1(\mathbf{q}_1^1 + \mathbf{q}_2^1 + \mathbf{q}_3^1) + \mathbf{p}_1(\mathbf{q}_1^2 + \mathbf{q}_2^2 + \mathbf{q}_3^2) = \mathfrak{p}_1(\mathfrak{q}_2 + \mathfrak{q}_3 + \mathfrak{q}_4)$. Without losing generality, let us assume that $\mathbf{p}_1\mathbf{q}_1^1 \geq \mathbf{p}_1(\mathbf{q}_1^1 + \mathbf{q}_2^1 + \mathbf{q}_3^1)$.

A similar reasoning applies to $\mathfrak{p}_2\mathfrak{q}_2 \geq \mathfrak{p}_2(\mathfrak{q}_1 + \mathfrak{q}_3)$ and $\mathfrak{p}_3\mathfrak{q}_3 \geq \mathfrak{p}_3(\mathfrak{q}_1 + \mathfrak{q}_4)$. However, since $\mathbf{p}_1\mathbf{q}_1^1 \geq \mathbf{p}_1(\mathbf{q}_1^1+\mathbf{q}_2^1+\mathbf{q}_3^1)$, we can now conclude that it must be that $\mathbf{p}_2\mathbf{q}_2^2 \geq \mathbf{p}_2(\mathbf{q}_1^2+\mathbf{q}_3^2)$ and $\mathbf{p}_3\mathbf{q}_3^2 \geq \mathbf{p}_3(\mathbf{q}_1^2+\mathbf{q}_4^2)$. Indeed, otherwise we would have feasible personalized quantities that lead to a SARP rejection for $\ell = 1$. Note that these inequalities imply that $2R_0^2 3$ and $3R_0^2 4$, meaning that $2R^2 4$.

Finally, using the same argument once more for $\mathfrak{p}_4\mathfrak{q}_4 \geq \mathfrak{p}_4(\mathfrak{q}_1 + \mathfrak{q}_2)$, we can conclude that any specification of feasible personalized quantities leads to a rejection of 2-SARP. We always obtain a rejection of SARP for either $\ell = 1$ or $\ell = 2$. To finish the proof of part 1, we have to note that exactly the same reasoning holds if all goods are public (or even any intermediate case with both private and public goods).

To prove part 2 of the lemma, we need to give one specification of feasible personalized quantities that satisfies 2-WARP, and another specification of feasible personalized prices that satisfies 2-WARP.

For the private case, let us consider the specification of feasible personalized quantities in Table 5.3.

$\mathbf{q}_1^1 = (5, 0, 0)$	$\mathbf{q}_1^2 = (0, 0, 0)$
$\mathbf{q}_2^1 = (0, 0, 0)$	$\mathbf{q}_2^2 = (0, 5, 0)$
$\mathbf{q}_3^1 = (0, 0, 0)$	$\mathbf{q}_3^2 = (0, 0, 5)$
$\mathbf{q}_4^1 = (0, 0, 0)$	$\mathbf{q}_4^2 = (4, 3, 1)$

Table 5.3: *Values of* $\mathbf{q}_t^1$ *and* $\mathbf{q}_t^2$ *for* $t = 1, 2, 3, 4$.

For this specification we obtain the scalar products in Table 5.4, which allow us to conclude that 2-WARP is satisfied.

$\mathbf{p}_1\mathbf{q}_1^1 = 37.5$	$\mathbf{p}_2\mathbf{q}_1^1 = 5$	$\mathbf{p}_3\mathbf{q}_1^1 = 1$	$\mathbf{p}_4\mathbf{q}_1^1 = 5$
$\mathbf{p}_1\mathbf{q}_2^1 = 0$	$\mathbf{p}_2\mathbf{q}_2^1 = 0$	$\mathbf{p}_3\mathbf{q}_2^1 = 0$	$\mathbf{p}_4\mathbf{q}_2^1 = 0$
$\mathbf{p}_1\mathbf{q}_3^1 = 0$	$\mathbf{p}_2\mathbf{q}_3^1 = 0$	$\mathbf{p}_3\mathbf{q}_3^1 = 0$	$\mathbf{p}_4\mathbf{q}_3^1 = 0$
$\mathbf{p}_1\mathbf{q}_4^1 = 0$	$\mathbf{p}_2\mathbf{q}_4^1 = 0$	$\mathbf{p}_3\mathbf{q}_4^1 = 0$	$\mathbf{p}_4\mathbf{q}_4^1 = 0$
$\mathbf{p}_1\mathbf{q}_1^2 = 0$	$\mathbf{p}_2\mathbf{q}_1^2 = 0$	$\mathbf{p}_3\mathbf{q}_1^2 = 0$	$\mathbf{p}_4\mathbf{q}_1^2 = 0$
$\mathbf{p}_1\mathbf{q}_2^2 = 2.5$	$\mathbf{p}_2\mathbf{q}_2^2 = 10$	$\mathbf{p}_3\mathbf{q}_2^2 = 10.1$	$\mathbf{p}_4\mathbf{q}_2^2 = 5$
$\mathbf{p}_1\mathbf{q}_3^2 = 2.5$	$\mathbf{p}_2\mathbf{q}_3^2 = 4.5$	$\mathbf{p}_3\mathbf{q}_3^2 = 10$	$\mathbf{p}_4\mathbf{q}_3^2 = 20$
$\mathbf{p}_1\mathbf{q}_4^2 = 32$	$\mathbf{p}_2\mathbf{q}_4^2 = 10.9$	$\mathbf{p}_3\mathbf{q}_4^2 = 8.86$	$\mathbf{p}_4\mathbf{q}_4^2 = 11$

Table 5.4: *Scalar products for the private case.*

Similarly, for the public case, we consider the specification of feasible personalized prices in Table 5.5.

For this specification we obtain the scalar products in Table 5.6, which allow us to conclude that 2-WARP is satisfied.

□

The next non-equivalence conclusion follows directly from Lemma 5.3.1.

$\mathbf{P}_1^1 = (7.5, 0, 0)$	$\mathbf{P}_1^2 = (0, 0.5, 0.5)$
$\mathbf{P}_2^1 = (0.2, 0.1, 0.2)$	$\mathbf{P}_2^2 = (0.8, 1.9, 0.7)$
$\mathbf{P}_3^1 = (0.2, 0.1, 0.1)$	$\mathbf{P}_3^2 = (0, 1.92, 1.9)$
$\mathbf{P}_4^1 = (1, 0, 0)$	$\mathbf{P}_4^2 = (0, 1, 5)$

Table 5.5: *Values of* $\mathbf{P}_t^1$ *and* $\mathbf{P}_t^2$ *for* $t = 1, 2, 3, 4$.

$\mathbf{P}_1^1\mathbf{Q}_1 = 37.5$	$\mathbf{P}_2^1\mathbf{Q}_1 = 1$	$\mathbf{P}_3^1\mathbf{Q}_1 = 1$	$\mathbf{P}_4^1\mathbf{Q}_1 = 5$
$\mathbf{P}_1^1\mathbf{Q}_2 = 0$	$\mathbf{P}_2^1\mathbf{Q}_2 = 0.5$	$\mathbf{P}_3^1\mathbf{Q}_2 = 0.5$	$\mathbf{P}_4^1\mathbf{Q}_2 = 0$
$\mathbf{P}_1^1\mathbf{Q}_3 = 0$	$\mathbf{P}_2^1\mathbf{Q}_3 = 1$	$\mathbf{P}_3^1\mathbf{Q}_3 = 0.5$	$\mathbf{P}_4^1\mathbf{Q}_3 = 0$
$\mathbf{P}_1^1\mathbf{Q}_4 = 30$	$\mathbf{P}_2^1\mathbf{Q}_4 = 1.3$	$\mathbf{P}_3^1\mathbf{Q}_4 = 1.2$	$\mathbf{P}_4^1 Q_4 = 4$
$\mathbf{P}_1^2\mathbf{Q}_1 = 0$	$\mathbf{P}_2^2\mathbf{Q}_1 = 4$	$\mathbf{P}_3^2\mathbf{Q}_1 = 0$	$\mathbf{P}_4^2\mathbf{Q}_1 = 0$
$\mathbf{P}_1^2\mathbf{Q}_2 = 2.5$	$\mathbf{P}_2^2\mathbf{Q}_2 = 9.5$	$\mathbf{P}_3^2\mathbf{Q}_2 = 9.6$	$\mathbf{P}_4^2\mathbf{Q}_2 = 5$
$\mathbf{P}_1^2\mathbf{Q}_3 = 2.5$	$\mathbf{P}_2^2\mathbf{Q}_3 = 3.5$	$\mathbf{P}_3^2\mathbf{Q}_3 = 9.5$	$\mathbf{P}_4^2\mathbf{Q}_3 = 25$
$\mathbf{P}_1^2\mathbf{Q}_4 = 2$	$\mathbf{P}_2^2\mathbf{Q}_4 = 9.6$	$\mathbf{P}_3^2\mathbf{Q}_4 = 7.66$	$\mathbf{P}_4^2\mathbf{Q}_4 = 8$

Table 5.6: *Scalar products for the public case.*

Proposition 5.3.2. *There exists a dataset S with only 3 goods that satisfies* 2-WARP *but not* 2-SARP. *In general, this implies that* 2-WARP *is not equivalent to* 2-SARP *for $N + K = 3$. This non-equivalence conclusion is independent of the public or private nature of the goods.*

It is possible to construct datasets (similar to the one in Example 5.3.1) to obtain exactly the same conclusion in a setting with L household members and $L + 1$ goods. The following proposition states the general result.

Proposition 5.3.3. *Let $L \geq 2$. There exists a dataset S consisting of four observations, and with only $L+1$ goods, that satisfies L-*WARP *but not L-*SARP. *In general, this implies that L-*WARP *is not equivalent to L-*SARP *for $N + K = L + 1$. This non-equivalence conclusion is independent of the public or private nature of the goods.*

Proof. Take $L \geq 3$. We start with the dataset S from example 5.3.1 and we add goods $4, 5, \ldots, L + 1$ and observations $5, 6, \ldots, L + 2$ to it.

The quantities for goods $4, 5, \ldots, L+1$ are 0 in observations $1, 2, 3$ and 4 and the corresponding prices in these observations are $\epsilon < \frac{0.14}{L}$. It can be checked that in this case the following inequalities hold: $\mathfrak{p}_1\mathfrak{q}_1 > \mathfrak{p}_1(\mathfrak{q}_2+\ldots+\mathfrak{q}_{L+2}), \mathfrak{p}_2\mathfrak{q}_2 > \mathfrak{p}_2(\mathfrak{q}_1+\mathfrak{q}_3+\mathfrak{q}_5+\ldots+\mathfrak{q}_{L+2}), \mathfrak{p}_3\mathfrak{q}_3 > \mathfrak{p}_3(\mathfrak{q}_1+\mathfrak{q}_4+\mathfrak{q}_5+\ldots+\mathfrak{q}_{L+2})$ and $\mathfrak{p}_4\mathfrak{q}_4 > \mathfrak{p}_4(\mathfrak{q}_1 + \mathfrak{q}_2 + \mathfrak{q}_5 + \ldots + \mathfrak{q}_{L+2})$.

For each observation $t = 5, 6, \ldots, L+2$, it is the case that $\mathfrak{q}_t = (0, \ldots, 0, 1, 0, \ldots, 0)$, with the non-zero quantity for the $(t-1)^{th}$ good. The corresponding prices are $\mathfrak{p}_t = (\epsilon, \ldots, \epsilon, 1, \epsilon, \ldots, \epsilon)$. Again, it can be checked that $\mathfrak{p}_t\mathfrak{q}_t > \mathfrak{p}_t(\mathfrak{q}_1 + \ldots + \mathfrak{q}_{t-1} + \mathfrak{q}_{t+1} + \mathfrak{q}_{L+2})$.

A similar reasoning as in Lemma 5.3.1 then shows that this dataset does not satisfy L-SARP. Personalized quantities or prices can be found as follows. Assign all goods (prices) of observations $1, 2, 3$ and 4 to members 1 and 2 as in the original examples. Furthermore, assign all goods (prices) of observation $k \geq 5$ to member $k-2$, $k = 5, \ldots, L+2$. □

5.3.3 Labour supply setting

Let us now turn to the restricted labour supply setting. More precisely, we consider a household with L members in which there is only private consumption of the $L+1$ goods. The first L goods represent leisure and are exclusively consumed by individual members. The $(L+1)$-th good is a Hicksian aggregate. We will treat this Hicksian good as a private good for which we do not observe the intrahousehold allocation. As indicated in Section 5.1, this restricted setting corresponds to the labour supply model of Chiappori (1988), which is widely used in empirical analyses of collective consumption behavior. Importantly, while we treat the Hicksian aggregate as a private good to facilitate our discussion (and for the analogy with Chiappori's original model), our following results actually also hold if the Hicksian aggregate were a public good.[1]

To formally explain the relation between this restricted setting and the gen-

[1] The proof of Proposition 5.3.4 is easily adapted.

eral setting that we discussed before, assume a dataset $S = \{(\mathbf{p}_t, 0; \mathbf{q}_t, 0), t = 1, ..., n\}$ that contains $L + 1$ private goods. Then the first L entries of the feasible personalized quantities $\mathbf{q}_t^\ell$ pertain to the exclusive goods, and are defined as follows for all $t = 1, \ldots, n$ and $\ell, j = 1, \ldots, L$:

$$\mathbf{q}_{t,\ell}^\ell = \mathbf{q}_{t,\ell}, \text{ and } \mathbf{q}_{t,j}^\ell = 0 \text{ if } \ell \neq j,$$

Thus, each member ℓ consumes only two goods: the exclusive ℓ-th good (of which the individual consumption is observed by construction) and a share of the $(L+1)$-th non-exclusive good (of which the individual consumption is not observed). Our proof of Proposition 4 exploits this two-goods feature. In particular, we can build on the original result of Rose (1958) to obtain the following conclusion.

Proposition 5.3.4. *Let $S = \{(\mathbf{p}_t, 0; \mathbf{q}_t, 0), t = 1, ..., n\}$ be a set of observations with $L+1$ goods. Assume households with L members of which each member consumes exclusively one of the goods. Then L-WARP is equivalent to L-SARP.*

Proof. Clearly, if S is a dataset that satisfies L-SARP, then it also needs to satisfy L-WARP. So we only need to prove the reverse statement. Let S be a dataset that satisfies L-WARP. This means that there exist feasible personalized quantities $\mathbf{q}_t^\ell$ such that for each $\ell = 1, \ldots, L$ the data $\{(\mathbf{p}_t, \mathbf{q}_t^\ell), t =, 1, \ldots, n\}$ satisfies WARP. By construction, all entries of $\mathbf{q}_t^\ell$ are zero except for the ℓ-th and $L+1$-th entries. Clearly, all the zero entries are irrelevant for checking consistency with WARP. Therefore, we can use Rose's result to conclude that, for each member ℓ, the corresponding SARP condition is met, and thus that S satisfies L-SARP. □

Thus, if L (out of $L+1$) goods are exclusive, then transitivity of individual preferences does not have empirical bite. As a result, the empirical analysis of multi-member consumption behavior need not explicitly account for transitivity, which can substantially alleviate the computational burden in practical applications. For instance, Cherchye et al. (2011) introduced an

integer programming method to check consistency of a dataset S with revealed preference axioms of collective consumption models.[2] When using this method for the L-SARP condition that we consider here, the equivalence result in Proposition 5.3.4 implies that we can drop n^3 transitivity constraints without affecting the conclusions of the analysis. Given that integer programming is often time consuming, this may considerably facilitate the empirical analysis when n gets large.

5.4 Conclusion

We showed that, in general, the equivalence between WARP and SARP for 2 goods does not generalize to L-WARP and L-SARP for $L+1$ goods. The implication is that transitivity of preferences does have testable implications. By contrast, the equivalence between L-WARP and L-SARP does hold for $L+1$ goods if each of the L household members is the exclusive consumer of one good (as in the collective labour supply setting of Chiappori (1988)). In that case, transitivity does not generate empirical bite. This can substantially facilitate the empirical revealed preference analysis in practical applications.

[2] Actually, Cherchye et al. (2011) consider the *Generalized Axiom of Revealed Preferences* (GARP) rather than SARP in their analysis. However, the integer programming problem for L-SARP is directly analogous to the one based on GARP. For compactness, we do not include it here.

Chapter 6

The Weak Axiom of Revealed Preference for Collective Households

6.1 Introduction

In the previous chapter, we investigated the possibility of equivalence between collective models with (L-SARP) and without (L-WARP) transitivity. Our main findings is that transitivity does have testable implications in collective models, as soon as there are more goods than household members (If there exist fewer goods, both models are trivially satisfied). However, it turns out that in practical applications they often have identical empirical implications, i.e. most data that satisfy WARP also satisfy SARP. Putting it differently, in empirical work transitivity usually plays little role when testing data consistency with revealed preference axioms. This observation is an important one in view of practical tests of the collective models, as tests of collective versions of SARP are known to be difficult to test. Most

This chapter is the result of a collaboration with Laurens Cherchye, Bram De Rock, Fabrice Talla Nobibon and Frits C.R. Spieksma. An article based on this chapter has been submitted for publication.

notably, it has been shown that testing the SARP conditions for collective models is NP-COMPLETE, even for households with only two members Deb (2010); Talla Nobibon et al. (2013); Talla Nobibon and Spieksma (2010).

This directly motivates the purpose of the current chapter, which focuses on the computational complexity of the collective WARP conditions. Essentially, we will evaluate whether the computational hardness of the collective revealed preference conditions can be mitigated by dropping the transitivity requirement. In particular, our following analysis will consider the WARP characterization of three collective consumption settings: (i) the *private* setting where all goods are consumed privately without externalities, (ii) the *public* setting, where all goods are publicly consumed inside the household, and (iii) a *general* setting where no information on the (private or public nature) of the goods is available.

Our main findings can be summarized as follows. A first "negative" conclusion will be that testing the collective WARP conditions is computationally difficult (i.e. NP-COMPLETE) for the private and public settings. In these cases, dropping transitivity does not solve the hardness problem associated with the collective SARP conditions. However, as a second "positive" conclusion, we also show that testing collective WARP for two members is computationally easy for the general setting. Here, we can effectively test consistency with the collective consumption model in an efficient way (i.e. in polynomial time) if we omit transitivity. (As we will indicate, for this general setting the complexity in the case of three or more member remains an open question.)

The remainder of the chapter unfolds as follows. Section 6.2 presents our basic set-up. Sections 6.3, 6.4 and 6.5 contain our main complexity results (for, respectively, the private, public and general settings). Section 6.6 concludes.

6.2 Notation

We consider multi-member households that take consumption decisions over m goods. These goods can be consumed either privately (with or without externalities) or publicly. More precisely, *private* consumption of a good means that the consumption by one household member affects the supply available for the other household members (e.g. drinking water can only be consumed privately). Next, consumption *externalities* refer to the fact that one household member gets utility from another household member's private consumption (e.g. a wife enjoys her husband's nice clothes). Finally, *public* consumption of a good means that consumption of that good by one household member does not affect the supply available for the other household members, and no one can be excluded from consuming the good (e.g. the rent of a shared house represents public consumption).

The collective models of household consumption explicitly recognize the individual preferences of the household members. These preferences may depend on the private quantities (with or without externalities), the public quantities, or both. Throughout, we assume that preferences of the household members can be represented by a well-behaved (i.e. continuous, positive monotonic and concave) utility function. The following sections will define explicit specifications of these member-specific utility functions for alternative collective consumption models.

We assume a setting in which the empirical analyst observes n household decisions resulting in consumption quantity bundles $q_i := (q_{i,1}, \ldots, q_{i,m}) \in \mathbb{R}^m_+$, with corresponding prices $p_t := (p_{t,1}, \ldots, p_{t,m}) \in \mathbb{R}^m_{++}$, $t = 1, \ldots, n$. The component $q_{t,j}$ (respectively $p_{t,j}$), for $j = 1, \ldots, m$, corresponds to the quantity of good j bought by the household (respectively, the unit price of good j) at the time of observation t. Note that the pq represents the total cost of the bundle $q \in \mathbb{R}^m_+$ at the prices $p \in \mathbb{R}^m_{++}$. We denote the set of observations by $S := \{(p_t, q_t) : \ t \in N\}$, where $N := \{1, \ldots, n\}$, and we refer to S as the *dataset*. For ease of exposition, throughout this chapter, we use $t \in N$ to refer to the observation (p_t, q_t).

6.3 The collective consumption model with only private consumption and no externalities

In the first collective consumption model that we study, we assume that all goods are consumed privately without externalities. In other words, the member-specific utility functions only depend on the private goods consumed by that member. To facilitate our discussion, we will mainly focus on two-member households in what follows. However, as we will also indicate in Theorem 2, our NP-COMPLETENESS result for two-member households can easily be generalized to households with L members ($L \geq 2$).

Because a typical dataset only contains information on consumption quantities that apply to the aggregate household level, we have to deal with the fact that we do not know which fraction of the observed bundle q_t is consumed by each individual household member. To this end, we consider, for each observation $t \in N$, a *feasible personalized quantity vector* (q_t^1, q_t^2), which describes the division of the goods over the two household members. Since the true split up of q_t is unobserved, we clearly need to consider all possible feasible personalized quantity vectors.[1] For each member ℓ ($\ell = 1, 2$) we define the *personalized consumption dataset* by $S_\ell = \{(p_t, q_t^\ell) : t \in N\}$.

The extension of WARP to this collective consumption model is then as follows.

Definition 6.3.1. PRIVATE 2-WARP
Let $S = \{(p_t, q_t) : t \in N\}$ be a dataset of a two-member household. We say that S is consistent with private 2-WARP *if and only if:*

(i) *For each $t \in N$ there exist q_t^1, $q_t^2 \in \mathbb{R}^m_+$ such that $q_t = q_t^1 + q_t^2$, and*

[1] In some datasets we have some information on how consumption is shared. This is called assignable information or exclusive goods in the literature. Such information can easily be integrated in our analysis and would not change our results.

(ii) *For each member* $\ell \in \{1, 2\}$*, the set* $S_\ell = \{(p_t, q_t^\ell) : t \in N\}$ *satisfies* WARP.

This problem can be rephrased as the following decision problem:

Problem 6.3.1. PRIVATE 2-WARP
Instance: A dataset $S = \{(p_t, q_t) : t \in N\}$.
Question: Do there exist q_t^1, $q_t^2 \in \mathbb{R}^m_+$ *satisfying* $q_t = q_t^1 + q_t^2$ *for each* $t \in N$ *such that for* $\ell = 1, 2$*, the set* $S_\ell = \{(p_t, q_t^\ell) : t \in N\}$ *satisfies* WARP*?*

It turns out that answering this question is NP-complete.

Theorem 6.3.1. *Testing* PRIVATE 2-WARP *is* NP-COMPLETE.

Proof. We use a reduction from MONOTONE NOT-ALL-EQUAL 3-SAT, which is known to be NP-COMPLETE (Garey and Johnson, 1979) and is defined as follows.

Problem 6.3.2. MONOTONE NOT-ALL-EQUAL 3-SAT
Instance: A set of variables $X = \{x_1, x_2, \ldots, x_n\}$ *and a set of clauses* $C = \{c_1, c_2, \ldots, c_m\}$ *with each clause consisting of 3 non-negated literals.*
Question: Does there exist a truth-assignment so that for each clause, either one or two of the literals are TRUE*?*

It is not difficult to see that PRIVATE 2-WARP belongs to the class NP. The rest of this proof is structured as follows: given an arbitrary instance of MNAE 3-SAT, we first build an instance of PRIVATE 2-WARP and next, we prove that we have a YES instance of MNAE 3-SAT if and only if the constructed instance of PRIVATE 2-WARP is a YES instance.

Consider an arbitrary instance $X = \{x_1, x_2, \ldots, x_n\}$ and $C = \{c_1, c_2, \ldots, c_m\}$ of MNAE 3-SAT. We build an instance of private 2-WARP using $3n+4$ goods and $2n+2m+3$ observations. We next describe the quantity and the price of goods for each observation. We use $\epsilon = \frac{1}{4n}$ and $M = n + 1$. The first

block of $2n$ observations corresponds to the variables and is given in table 6.1

The second block of $2m$ observations corresponds to the clauses. For each clause $c_a = \{x_i, x_j, x_k\}$, we have the observations $2n + a$ and $2n + m + a$ $(a = 1, \ldots, m)$. The prices and quantities are given in table 6.2. The prices of the goods corresponding to variables x_i, x_j and x_k equal 1, and the prices of the goods corresponding to other variables equal ϵ. Finally, we have observations $2n+2m+1$, $2n+2m+2$, $2n+2m+3$, given in table 6.3.

We have now described the dataset S. Before embarking further on the proof, let us describe the main idea. Consider the n goods, $5, 6, 7, \ldots, n+4$ in observation $2n + 2m + 3$. Each of these goods corresponds to a variable in the instance of MNAE-3SAT. We will argue that each of these n goods is allocated for a large part (i.e. $\geq \frac{3}{4}$) to some member $\ell \in \{1, 2\}$. This is akin to setting the corresponding variable to TRUE (if the good goes for the larger part to member 1), or to FALSE (if the good goes for the larger part to member 2). Of course it remains to show that this is a satisfying truth assignment. Recall that we say that for member $\ell \in \{1, 2\}$ q_a^ℓ is directly revealed preferred to q_b^ℓ, when we have $p_a q_a^\ell \geq p_a q_b^\ell$ with $a, b \in S$.

Claim 6.3.1. *If $p_a q_a \geq p_a q_b$ for some $a, b \in S, a \neq b$, then there exists an $\ell \in \{1, 2\}$ for which q_a^ℓ is directly revealed preferred to q_b^ℓ*

Proof. Consider any split of q_a into q_a^1, q_a^2, and q_b into q_b^1, q_b^2, i.e., let $q_a^1 + q_a^2 = q_a$ and $q_b^1 + q_b^2 = q_b$. Since $p_a q_a \geq p_a q_b$, it follows that $p_a(q_a^1 + q_a^2) \geq p_a(q_b^1 + q_b^2)$. Hence, either $p_a q_a^1 \geq p_a q_b^1$ or $p_a q_a^2 \geq p_a q_b^2$ (or both). □

Notice that, apart from bundle $q_{2n+2m+3}$, all other bundles are unit vectors. We will use $q_{i,j}$ $(p_{i,j})$ to denote the quantity (price) of good j in observation i, $i = 1, \ldots, 2n + 2m + 3, j = 1, \ldots, 3n + 4$. We now exhibit a trick that we will use throughout the proof. Consider a hypothetical dataset, containing the observations a and b as follows:

Table 6.1: *Observations corresponding to variables*

q_1	$= (0,0,\|0,0,\|0,\ldots,0,\|1,0,\ldots,0,\|0,0,\ldots,0)$	p_1	$= (M,M,\|M,M,\|1,\epsilon,\ldots,\epsilon,\|1,M,\ldots,M,\|\epsilon,M,\ldots,M)$
q_2	$= (0,0,\|0,0,\|0,\ldots,0,\|0,1,\ldots,0,\|0,0,\ldots,0)$	p_2	$= (M,M,\|M,M,\|\epsilon,1,\ldots,\epsilon,\|M,1,\ldots,M,\|M,\epsilon,\ldots,M)$
$\vdots$		$\vdots$	
q_n	$= (0,0,\|0,0,\|0,\ldots,0,\|0,0,\ldots,1,\|0,0,\ldots,0)$	p_n	$= (M,M,\|M,M,\|\epsilon,\epsilon,\ldots,1,\|M,M,\ldots,1,\|M,M,\ldots,\epsilon)$
q_{n+1}	$= (0,0,\|0,0,\|0,\ldots,0,\|0,0,\ldots,0,\|1,0,\ldots,0)$	p_{n+1}	$= (M,M,\|M,M,\|M,\ldots,M,\|\epsilon,M,\ldots,M,\|1,M,\ldots,M)$
$\vdots$		$\vdots$	
q_{2n}	$= (0,0,\|0,0,\|0,\ldots,0,\|0,0,\ldots,0,\|0,\ldots,0,1)$	p_{2n}	$= (M,M,\|M,M,\|M,\ldots,M,\|M,M,\ldots,\epsilon,\|M,M,\ldots,1)$

Table 6.2: *Observations corresponding to clauses*

q_{2n+a}	$= (1,0,\|0,0,\|0,\ldots,0,\|0,\ldots,0,\|0,\ldots,0)$	p_{2n+a}	$= (1,\epsilon,\|M,M,\|\{\epsilon,1\}\|M,\ldots,M,\|M,\ldots,M)$
q_{2n+m+a}	$= (0,1,\|0,0,\|0,\ldots,0,\|0,\ldots,0,\|0,\ldots,0)$	p_{2n+m+a}	$= (\epsilon,1,\|M,M,\|\{\epsilon,1\}\|M,\ldots,M,\|M,\ldots,M)$

Table 6.3: *Extra observations*

$q_{2n+2m+1}$	$= (0,0,\|1,0,\|0,\ldots,0,\|0,\ldots,0,\|0,\ldots,0)$	$p_{2n+2m+1}$	$= M,M,\|1,\epsilon,\|\epsilon,\ldots,\epsilon,\|M,\ldots,M,\|M,\ldots,M)$
$q_{2n+2m+2}$	$= 0,0,\|0,1,\|0,\ldots,0,\|0,\ldots,0,\|0,\ldots,0)$	$p_{2n+2m+2}$	$= (M,M,\|\epsilon,1,\|\epsilon,\ldots,\epsilon,\|M,\ldots,M,\|M,\ldots,M)$
$q_{2n+2m+3}$	$= (0,0,\|0,0,\|1,\ldots,1,\|0,\ldots,0,\|0,\ldots,0)$	$p_{2n+2m+3}$	$= (\epsilon,\epsilon,\|n-1,n-1,\|1,\ldots,1,\|\epsilon,\ldots,\epsilon,\|\epsilon,\ldots,\epsilon)$

$$q_a = (1, 0), \qquad p_a = (1, \epsilon)$$
$$q_b = (0, 1), \qquad p_b = (\epsilon, 1)$$

We say that the split of a bundle is *extreme* if each unit good of that bundle goes to one (of the two) members with fraction at least $1 - \epsilon$.

Claim 6.3.2. *In any feasible solution to* PRIVATE 2-WARP *of some dataset containing observations a and b, the split of bundles q_a and q_b is extreme.*

Proof. Clearly, we have both $p_a q_a > p_a q_b$ and $p_b q_b > p_b q_a$. So using Claim 1, it follows that for one member ℓ we have that q_a^ℓ is directly revealed preferred to q_b^ℓ and simultaneously for one member ℓ' $q_b^{\ell'}$ is directly revealed preferred to $q_a^{\ell'}$. Thus in any feasible solution ℓ and ℓ' must be different (otherwise private 2-WARP is violated). Let us assume, without loss of generality, that for member 1 we have that q_a^1 is directly revealed preferred to q_b^1 and q_b^1 is not directly revealed preferred to q_a^1. Let α be the fraction of bundle a allocated to member 1, and β the fraction of bundle b allocated to this member. We then have that

$$p_b q_b^1 < p_b q_a^1 \Rightarrow \beta < \epsilon\alpha.$$

Since $\alpha \leq 1$, we conclude $\beta < \epsilon = \frac{1}{4n}$. Likewise, since for member 2 we have that q_a^2 is not directly revealed preferred to q_b^2, we find:

$$p_a q_a^2 < p_a q_b^2 \Rightarrow 1 - \alpha < \epsilon(1 - \beta) \Rightarrow \alpha > 1 - \epsilon = \frac{4n - 1}{4n}.$$

Claim 6.3.2 follows. □

Clearly, Claim 6.3.2 is applicable to any pair of observations involving bundles that are unit vectors, and price vectors that feature price ϵ and price 1. When applying this claim further on in our proof, some price vectors also include price M. However, whenever this is the case, the quantity of the goods will be 0 in both bundles, so the M prices can be ignored in

these situations.

We now proceed to show that when the constructed instance of private 2-WARP is a YES-instances, a satisfying truth assignment exists.

Claim 6.3.3. *In any feasible solution to this instance of private* 2-WARP*, we have for* $\ell = 1, 2$*:* $p_{2n+2m+3} q^{\ell}_{2n+2m+3} > 1$.

Proof. Observe that Claim 6.3.2 is applicable to observations $2n + 2m + 1$ and $2n + 2m + 2$. Thus the split of the bundles $q_{2n+2m+1}$ and $q_{2n+2m+2}$ is extreme. Let us assume, without loss of generality, that good 3 is allocated to member 1 with fraction at least $1-\epsilon$, while good 4 is allocated to member 2 with fraction $1 - \epsilon$. Thus:

$$\begin{aligned} p_{2n+2m+1} q^1_{2n+2m+1} \geq 1 - \epsilon = \frac{4n-1}{4n} \geq \frac{1}{4} \\ = n\frac{1}{4n} = n\epsilon \geq p_{2n+2m+1} q_{2n+2m+3}. \end{aligned} \quad (6.1)$$

It follows that for member 1, in any feasible solution, observation $q^1_{2n+2m+1}$ is revealed preferred over $q^1_{2n+2m+3}$. Then, in order to satisfy private 2-WARP, we must have:

$$\begin{aligned} p_{2n+2m+3} q^1_{2n+2m+3} < p_{2n+2m+3} q^1_{2n+2m+1} \leq p_{2n+2m+3} q_{2n+2m+1} \\ \Rightarrow \sum_{i=1}^{n} q^1_{2n+2m+3,4+i} < n - 1. \end{aligned} \quad (6.2)$$

Since, for $a \in S$, $q^2_a = q_a - q^1_a$, we derive, using (6.2):

$$\sum_{i=1}^{n} q^2_{2n+2m+3,4+i} = n - \sum_{i=1}^{n} q^1_{2n+2m+3,4+i} > 1. \quad (6.3)$$

Finally, since $p_{2n+2m+3,i} = 1$ for $i = 5, 6, \ldots, n + 4$, it follows that (6.3) can be written as:

$$p_{2n+2m+3} q^2_{2n+2m+3} > 1.$$

A similar reasoning involving member 2 and observations $2n+2m+2$ and $2n+2m+3$ leads to:

$$p_{2n+2m+3}q^1_{2n+2m+3} > 1.$$

□

To proceed, let us consider observation i, $(1 \leq i \leq n)$, and observation $2n+2m+3$. Using Claim 3, we observe:

$$p_{2n+2m+3}q^\ell_{2n+2m+3} > 1 > p_{2n+2m+3}q_i \qquad \text{for } \ell = 1,2, 1 \leq i \leq n. \tag{6.4}$$

Thus, no matter the split of q_i into q^1_i and q^2_i, for both members 1 and member 2 we have that $q^1_{2n+2m+3}$ (resp. $q^2_{2n+2m+3}$) is directly revealed preferred over observation q^1_i (resp. q^2_i), with $i = 1, \ldots, n$. Since we have a YES-instance of 2-WARP, we know that then, for $\ell = 1, 2$:

$$p_i q^\ell_i < p_i q^\ell_{2n+2m+3}. \tag{6.5}$$

Observe that Claim 6.3.2 is applicable to observations i and $n+i$. Thus, the split of q_i and q_{n+i} is extreme. Hence, there is a member ℓ for which:

$$p_i q^\ell_i \geq 1 - \epsilon. \tag{6.6}$$

Inequalities (6.5) and (6.6) imply that the split of $q_{2n+2m+3}$ is such that:

$$p_i q^\ell_{2n+2m+3} > 1 - \epsilon. \tag{6.7}$$

Consider the vectors p_i and $q_{2n+2m+3}$, $1 \leq i \leq n$. It follows that:

$$p_i q^\ell_{2n+2m+3} = \epsilon \sum_{j=5, j\neq 4+i}^{n+4} q^\ell_{2n+2m+3,j} + q^\ell_{2n+2m+3,4+i}. \tag{6.8}$$

Also:

$$\sum_{j=5,j\neq 4+i}^{n+4} q^{\ell}_{2n+2m+3,j} \leq \sum_{j=5,j\neq 4+i}^{n+4} q_{2n+2m+3,j} = n-1. \tag{6.9}$$

Rewriting (6.8), and using inequalities (6.7) and (6.9) gives for each $i = 1,\ldots,n$:

$$\begin{aligned} q^{\ell}_{2n+2m+3,4+i} &= p_i q^{\ell}_{2n+2m+3} - \epsilon \sum_{j=5,j\neq 4+i}^{n+4} q^{\ell}_{2n+2m+3,j} \\ &> 1-\epsilon-\epsilon(n-1) = 1-\frac{n}{4n} = \frac{3}{4}. \end{aligned} \tag{6.10}$$

Concluding, each good $i = 5,6,\ldots,n+4$ in observation $2n+2m+3$ is allocated for over $\frac{3}{4}$ to some member $\ell \in \{1,2\}$.

Finally, we look at the two observations corresponding to each clause $j = 1,\ldots,m$. It is clear that for each member ℓ we have that $q^{\ell}_{2n+2m+3}$ is directly revealed preferred over both observations q^{ℓ}_{2n+j} and q^{ℓ}_{2n+m+j}. Observe also that Claim 2 is applicable to observations $2n+j$ and $2n+2m+j$. Thus, in order not to have a violation of private 2-WARP, we should have for member ℓ q^{ℓ}_{2n+j} is not directly revealed preferred over $q^{\ell}_{2n+2m+3}$. Thus, for each $\ell = 1,2$:

$$p_{2n+j} q^{\ell}_{2n+j} < p_{2n+j} q^{\ell}_{2n+2m+3}. \tag{6.11}$$

Since (without loss of generality), for member 1, we have $p_{2n+j} q^{1}_{2n+j} \geq 1-\epsilon$, and thus we have using (6.11):

$$p_{2n+j} q^{1}_{2n+2m+3} > 1-\epsilon. \tag{6.12}$$

This means that one of the three goods associated to clause j is allocated over $\frac{3}{4}$ to member 1. We argue by contradiction. Indeed, in case none of the three goods of clause j are allocated over $\frac{3}{4}$ to member 1, then they are allocated for at most $\frac{1}{4}$ to member 1. Then,

$$p_{2n+j}q^1_{2n+2m+3} \leq 3\frac{1}{4} + (n-3)\epsilon = \frac{3}{4} + \frac{n-3}{4n} = \frac{4n-3}{4n} < \frac{4n-1}{4n}. \quad (6.13)$$

Thus we would have $p_{2n+j}q^1_{2n+2m+3} < 1-\epsilon$, contradicting (6.12). Therefore, at least one of the goods associated with j is allocated over $\frac{3}{4}$ to member 1. Clearly, a similar reasoning involving $2n+2m+j$ and member 2 implies that one of these three goods must be allocated over $\frac{3}{4}$ to member 2.

In conclusion, we now know the following about any valid allocation of observation $2n+2m+3$ which satisfies PRIVATE 2-WARP. First, that each good is split up in a large and a small allocation for the different members. Secondly, that for each clause and each member, there is at least one of the goods associated with the variables that has a large allocation. A valid truth assignment for MNAE 3-SAT can now be found as follows. If, in observations $2n+2m+3$ a good is largely allocated to member 1, the variable is set to TRUE, if a good is largely allocated to member 2, the variable is FALSE.

If we have a Yes-instance of MNAE 3-SAT, an allocation of goods which satisfies PRIVATE 2-WARP exists. For observation $2n+2m+3$, fully assign each good associated with a TRUE variable to member 1, and each good associated with a FALSE variable to member 2. Likewise, fully assign the bundle i to member 1 if x_i is TRUE and to member 2 if it is FALSE. Furthermore, for all $j = 1, \ldots, m$, fully assign bundles $2n+j$ to member 1 and all $2n+m+j$ to member 2. Finally, fully assign $2n+2m+1$ to member 1 and $2n+2m+2$ to member 2. It can be easily checked that such an

allocation satisfies PRIVATE 2-WARP. □

The following argument generalizes our NP-completeness results for PRIVATE 2-WARP towards private L-WARP for any fixed $L \geq 2$.[2]

Theorem 6.3.2. *Testing private L-WARP is NP-COMPLETE for any fixed $L \geq 2$.*

Proof. NP-Completeness for private L-WARP, $L > 2$ can also be proven through a reduction from MONOTONE NOT-ALL-EQUAL 3-SAT. We briefly sketch this reduction. The dataset constructed is the same as for $L = 2$, except that for any additional member beyond the second, one extra observation and one extra good is added. There are now $2n+2m+3+(L-2)$ observations and $3n + 4 + (L - 2)$ goods. Observation $2n + 2m + 3 + i$ consists of only one unit of good $3n + 4 + i$, which has price 1 and all other prices equal to ϵ. In all other observations, good $3n + 4 + i$ has price ϵ. Using an argument similar to the proof of Claim 6.3.2, it is clear that, in any feasible solution, allocations are extreme, i.e., any member ℓ who is allocated more than a small fraction of good $3n + 4 + i$ will prefer $q^{\ell}_{2n+2m+3+i}$ over all other bundles and any member who is allocated more than a fraction of a bundle in other observations prefers that bundle over $q_{2n+2m+3+i}$. Any feasible split will thus have the goods $3n + 4 + i$ in the extra observations almost completely allocated to the members $3, 4, \ldots$, while all of the bundles present in the proof for $L = 2$ must still be split over two members. □

[2] The definition of private L-WARP is trivially analogous to the one of PRIVATE 2-WARP. For compactness, we do not include it here.

6.4 The collective consumption model with only public goods

We next turn to the collective model with all goods publicly consumed in the household. In this case all member-specific utility functions are defined for the same bundle of public goods. For ease of exposition, we again mainly focus on households consisting of two members. But, like before, our findings for this case are easily extended to L-member households (with $L \geq 2$).

For this public setting, we formalize the idea that household members have individual-specific (unobserved) willingness-to-pay for the public goods (bought at prices p_t). To do so, for each observation $t \in N$, we define a *feasible personalized price vector* (p_t^1, p_t^2). Intuitively, these feasible personalized prices capture the fractions of the household prices for the public goods that are borne by the individual members ℓ. Given the Pareto efficiency assumption that underlies the collective consumption model, these prices can also be interpreted as Lindahl prices. We refer to Cherchye et al. (2011) for a detailed discussion.

Similar to before, for each member ℓ ($\ell = 1, 2$) we consider *personalized consumption datasets* by $S_\ell = \{(p_t^\ell, q_t) : t \in N\}$. The extension of WARP to this collective consumption model is then as follows.

Definition 6.4.1. PUBLIC 2-WARP
Let $S = \{(p_t, q_t) : t \in N\}$ be a dataset of a two-member household. We say that S is consistent with public 2-WARP *if and only if:*

(i) *For each $t \in N$ there exist p_t^1, $p_t^2 \in \mathbb{R}^m_+$ such that $p_t = p_t^1 + p_t^2$, and*

(ii) *For each member $\ell \in \{1, 2\}$, the set $S_\ell = \{(p_t^\ell, q_t) : t \in N\}$ satisfies* WARP.

This problem can be rephrased as the following decision problem:

Problem 6.4.1. PUBLIC 2-WARP
Instance: A dataset $S = \{(p_t, q_t) : t \in N\}$.
Question: Do there exist p_t^1, $p_t^2 \in \mathbb{R}_+^m$ *satisfying* $p_t = p_t^1 + p_t^2$ *for each* $t \in N$ *such that for* $\ell = 1, 2$, *the set* $S_\ell = \{(p_t^\ell, q_t) : t \in N\}$ *satisfies* WARP*?*

It turns out that answering this question also implies solving an NP-complete problem.

Theorem 6.4.1. *Testing* PUBLIC 2-WARP *is* NP-COMPLETE.

Proof. We again use a reduction from MONOTONE NOT-ALL-EQUAL 3-SAT, as defined by problem 6.3.2
Instance: A set of variables $X = \{x_1, x_2, \ldots, x_n\}$ and a set of clauses $C = \{c_1, c_2, \ldots, c_m\}$ with each clause consisting of 3 non-negated literals.
Question: Does there exist a truth-assignment so that for each clause, either one or two of the literals are TRUE?

It is not difficult to see that PUBLIC 2-WARP belongs to the class NP. The rest of this proof is structured as follows: given an arbitrary instance of MNAE 3-SAT, we first build an instance of PUBLIC 2-WARP and next, we prove that we have a YES instance of MNAE 3-SAT if and only if the constructed instance of PUBLIC 2-WARP is a YES instance.

Consider an arbitrary instance $X = \{x_1, x_2, \ldots, x_n\}$ and $C = \{c_1, c_2, \ldots, c_m\}$ of MNAE 3-SAT. We build an instance of public 2-WARP using $3n+2$ goods and $2n + m + 2$ observations. We next describe the quantity and the price of the goods for each observation. We use $\epsilon = \frac{1}{4n}$ and $M = n + 1$. The first block of $2n$ observations corresponds to the variables and is given by:

$$
\begin{array}{lll}
q_1 & = (2,0,\ldots,0,|0,\ldots,0,|1,0,\ldots,0,|0,0); & p_1 = (1,\epsilon,\ldots,\epsilon,|\epsilon,\ldots,\epsilon,|\epsilon,M,\ldots,M,|M,M) \\
q_2 & = (0,2,\ldots,0,|0,\ldots,0,|0,1,\ldots,0,|0,0); & p_2 = (\epsilon,1,\ldots,\epsilon,|\epsilon,\ldots,\epsilon,|M,\epsilon,\ldots,M,|M,M) \\
\vdots & & \\
q_n & = (0,0,\ldots,2,|0,\ldots,0,|0,0,\ldots,1,|0,0); & p_n = (\epsilon,\epsilon,\ldots,1,|\epsilon,\ldots,\epsilon,|M,M,\ldots,\epsilon,|M,M) \\
q_{n+1} & = (0,\ldots,0,|1,0,\ldots,0,|1,0,\ldots,0,|0,0); & p_{n+1} = (\epsilon,\ldots,\epsilon,|1,\epsilon,\ldots,\epsilon,|\epsilon,M,\ldots,M,|M,M) \\
\vdots & & \\
q_{2n} & = (0,\ldots,0,|0,0,\ldots,1,|0,0,\ldots,1,|0,0); & p_{2n} = (\epsilon,\ldots,\epsilon,|\epsilon,\epsilon,\ldots,1,|M,M,\ldots,\epsilon,|M,M)
\end{array}
$$

Notice that each entry i in both a price-vector and a quantity-vector correspond to good i, $i = 1,\ldots,3n+2$. The second block of m observations corresponds to the clauses. For each clause $c_a = \{x_i, x_j, x_k\}$, we have the observation $2n + a$ $(a = 1,\ldots,m)$.

$$
\begin{array}{lll}
q_{2n+1} & = (\{0,1\}|0,\ldots,0|0,\ldots,0|0,0); & p_{2n+1} = (\{M,1\}|M,\ldots,M|\epsilon,\ldots,\epsilon|2,2) \\
q_{2n+2} & = (\{0,1\}|0,\ldots,0|0,\ldots,0|0,0); & p_{2n+2} = (\{M,1\}|M,\ldots,M|\epsilon,\ldots,\epsilon|2,2) \\
\vdots & & \\
q_{2n+m} & = (\{0,1\}|0,\ldots,0|0,\ldots,0|0,0); & p_{2n+m} = (\{M,1\}|M,\ldots,M|\epsilon,\ldots,\epsilon|2,2)
\end{array}
$$

In the observations corresponding to a given clause c_a, $a = 1,\ldots,m$, the quantity of good i is 1 if the variable x_i is part of the clause c_a, and 0 otherwise. As for the prices, the price of good i is 1 if variable x_i is in clause c_a, M otherwise. Finally, we have observations $2n + m + 1$ and $2n + m + 2$:

$$
\begin{array}{lll}
q_{2n+m+1} & = (0,\ldots,0,|0,\ldots,0,|0,\ldots,0|1,0); & p_{2n+m+1} = (\epsilon,\ldots,\epsilon,|\epsilon,\ldots,\epsilon,|M,\ldots,M|1,\epsilon) \\
q_{2n+m+2} & = (0,\ldots,0,|0,\ldots,0,|0,\ldots,0|0,1); & p_{2n+m+2} = (\epsilon,\ldots,\epsilon,|\epsilon,\ldots,\epsilon,|M,\ldots,M|\epsilon,1)
\end{array}
$$

This concludes the description of the instance of PUBLIC 2-WARP. The main idea used to argue the equivalence between MONOTONE NOT-ALL-EQUAL 3-SAT and PUBLIC 2-WARP is as follows. The first n goods, $1,\ldots,n$ represent the variables considered. We will argue that the (unit) prices of

each of these n goods in these first n observations is allocated for a large part (i.e. $\geq 1-\epsilon$) to some member $\ell \in \{1,2\}$. This corresponds to setting the variable to TRUE (if the price goes for the larger part to member 1), or to FALSE (if the price goes for the larger part to member 2). The proof will show that this is a satisfying truth assignment.

We first show that the trick used in the proof for private goods can be reused similarly in the context of public goods. Consider a hypothetical dataset, containing the observations a and b as follows:

$$\begin{aligned} q_a &= (1,0), & p_a &= (1,\epsilon) \\ q_b &= (0,1), & p_b &= (\epsilon,1) \end{aligned}$$

We say that the split of the prices of a bundle is *extreme* if the unit prices of goods present in the bundle are allocated to one of the two members with amount at least $1-\epsilon$.

Claim 6.4.1. *In any feasible solution to* PUBLIC 2-WARP *of some dataset containing observations a and b as above, the split of the price vector is extreme.*

Proof. Clearly, we have both $p_a q_a > p_a q_b$ and $p_b q_b > p_b q_a$. In this case, we have for one member ℓ q_a is directly revealed preferred over q_b, and for one member ℓ' q_b is directly revealed preferred over q_a. Thus in any feasible solution ℓ and ℓ' must be different (otherwise public 2-WARP is violated). Let us assume, without loss of generality, that for member 1 q_b is directly revealed preferred over q_a, while q_a is not directly revealed preferred over q_b. Let α be the part of the price of product 1 in bundle a allocated to member 1, and β the part of the price of product 2 allocated to this member.We then have

$$p_a^1 q_a < p_a^1 q_b \Rightarrow \alpha < \beta$$

Since $\beta \leq \epsilon$, we conclude $\alpha < \epsilon = \frac{1}{4n}$. This implies the price of the first good of a is allocated for more than $1 - \epsilon$ to member 2. By the same argument, the price of the second good in observation b is allocated for more than $1 - \epsilon$ to member 1. Claim 6.4.1 follows. □

Claim 6.4.2. *In any feasible solution, the decomposition of p_{2n+m+1} is such that for some member ℓ we have that q_{2n+m+1} is directly revealed preferred over all bundles q_{2n+i}, with $i = 1, \ldots, m$, while for the other member $\ell' \neq \ell$ q_{2n+m+2} is directly revealed preferred over all q_{2n+i}.*

Proof. It is clear that Claim 6.4.1 may be directly applied to observations $2n + m + 1$ and $2n + m + 2$. Without loss of generality, we assume that in any feasible solution, the price of good $3n + 1$ in observation $2n + m + 1$ is allocated almost completely to member 1, while the price of good $3n + 2$ in observations $2n + m + 2$ is allocated to member 2. It can easily be checked that $p^1_{2n+m+1} q_{2n+m+1} > 1 - \epsilon$, while $p^1_{2n+m+1} q_{2n+i} \leq 3\epsilon$ for all $i = 1, \ldots, m$. A similar analysis for member 2 proves the claim. □

This claim allows us to find a condition on the split of the prices in the observations associated with the clauses.

Claim 6.4.3. *In any feasible solution, for any observation $2n + i$ and any member ℓ, the split of the corresponding prices must be so that $1 < p^\ell_{2n+i} q_{2n+i} < 2$.*

Proof. By Claim 6.4.2, for any member ℓ, we have that either q_{2n+m+1} is directly revealed preferred to q_{2n+i} or q_{2n+m+2} is directly revealed preferred to q_{2n+i}. Without loss of generality, we assume q_{2n+m+1} is directly revealed preferred to q_{2n+i}. In any feasible solution, it is then the case that $p^\ell_{2n+i} q_{2n+i} < p^\ell_{2n+i} q_{2n+m+1}$. It can be easily checked that $p^\ell_{2n+i} q_{2n+m+1}$ is at most 2. As $p_{2n+i} q_{2n+i} = 3$ and $p^1_{2n+i} + p^2_{2n+i} = p_{2n+i}$, $p^\ell_{2n+i} q_{2n+i} > 1$ follows immediately. □

Consider now a pair of observations i and $n+i$, $i = 1, \ldots, n$. While these prices and quantities do not coincide with those in Claim 6.4.1 exactly, it can be easily seen that the split of price i in observation i is extreme. Next, notice that the member ℓ to whom more than $1-\epsilon$ of the price of i is allocated, will have $p_i^\ell q_i > p_i^\ell q_{2n+a}$, with $a = 1, \ldots, m$. This brings us to the following claim

Claim 6.4.4. *In any feasible solution to* PUBLIC 2-WARP*, if there exists some clause c_a, $a = 1, \ldots, m$ with variables x_i, x_j, x_k, it can not be the case that for some member ℓ, the prices of i, j and k in respectively observations i, j and k are allocated for more than $1-\epsilon$ to ℓ.*

Proof. We argue by contradiction. Suppose that in a feasible solution to PUBLIC 2-WARP some member ℓ is allocated almost completely the prices of goods i, j, k occurring in some clause c_a in respective observations i, j, k. The following inequalities follow: $p_i^\ell q_i > p_i^\ell q_{2n+a}$, $p_i^\ell q_i > p_i^\ell q_{2n+a}$, $p_i^\ell q_i > p_i^\ell q_{2n+a}$. Hence, in order not to violate WARP for member ℓ, we must have:

$$p_{2n+a}^\ell q_{2n+a} < p_{2n+a}^\ell q_i \tag{6.14}$$

$$p_{2n+a}^\ell q_{2n+a} < p_{2n+a}^\ell q_j \tag{6.15}$$

$$p_{2n+a}^\ell q_{2n+a} < p_{2n+a}^\ell q_k \tag{6.16}$$

As Claim 6.4.3 shows that $p_{2n+c}^\ell q_{2n+c} > 1$, we must have $p_{2n+c}^\ell q_i > 1$. This can be rewritten as $2 \times p_{2n+c,i}^\ell + p_{2n+c,2n+i}^\ell > 1$, for convenience, we will ignore $p_{2n+c,2n+i}^\ell$ as it is negligeable in the following analysis. We now have $p_{2n+c,i}^\ell > \frac{1}{2}$. However, this must hold for all three prices associated with i, j, k, which gives the following $p_{2n+c}^\ell q_{2n+c} = p_{2n+c,i}^\ell + p_{2n+c,j}^\ell + p_{2n+c,k}^\ell > \frac{3}{2}$. In this case, $p_{2n+c,i}^\ell > \frac{1}{2}$ is no longer sufficient, as $2 \times p_{2n+c,i}^\ell + p_{2n+c,2n+i}^\ell > \frac{3}{2}$ is required. By the same argument, we obtain that $p_{2n+c,i}^\ell > \frac{3}{4}$. However, in this case $p_{2n+c,i}^\ell + p_{2n+c,j}^\ell + p_{2n+c,k}^\ell > \frac{9}{4}$. However, by Claim 6.4.3 this can not be the case for a feasible solution. By contradiction, Claim 6.4.4 is thus proven. □

Now, it has become easy to show that a YES-instance of PUBLIC 2-WARP

problem corresponds to a satisfying truth assignment in MNAE-3-SAT, and vice versa. It is clear that - if a feasible solution to PUBLIC 2-WARP exists - the prices of good i in observation i, $i = 1, \ldots, n$ are always allocated with an extreme split. If the price of good i in observation i is almost completely allocated to member 1, we set the corresponding variable x_i to TRUE, otherwise we set it to FALSE. By Claim 6.4.4, we know that if a feasible solution to public WARP exists, and goods i, j, k are in a clause, no member will have the prices of all 3 goods allocated to him/her. Thus, a solution to PUBLIC 2-WARP corresponds to a satisfying truth assignment in MNAE-3-SAT. The other direction, i.e., finding a solution to PUBLIC 2-WARP when a satisfying truth assignment in MNAE-3-SAT is given is easy: simply allocate almost completely good i in observation i to member 1 if x_i is TRUE, else allocate good i almost completely to member 2, $i = 1, \ldots, n$. For observations $n + i$ and goods $n + 1$, the reverse is done. $2n + m + 1$ and $2n + m + 2$ are respectively allocated to member 1 and 2. All other prices may be split evenly between the members. This will satisfy PUBLIC 2-WARP. □

Similar to the private setting, we can extend our NP-COMPLETENESS results for PUBLIC 2-WARP to public L-WARP for any fixed $L \geq 2$.[3]

Theorem 6.4.2. *Testing public* L-WARP *is* NP-COMPLETE *for any fixed* $L \geq 2$.

Proof. The proof for this Theorem is analogous to the proof of Theorem 6.3.2. □

6.5 The general collective consumption model

The final collective consumption model that we consider is the most general one. It does not make any assumption regarding the nature of the

[3] Again, the definition of public L-WARP is directly analogous to the one of PUBLIC 2-WARP and, therefore, we do not include it here.

consumed goods. That is, every good can be privately or publicly consumed, and the private goods may generate externalities. If a dataset S is a YES-instance to either public or private 2-WARP, it is thus also a YES-instance to general 2-WARP. Clearly, the converse is not necessarily true. As before, we only observe data at the aggregate household level.

Differing from before, we no longer use the notions of feasible personalized prices and quantities to characterize this general collective model. Instead, we follow the approach developed in Cherchye et al. (2007, 2012), which defines a revealed preference characterization in terms of *hypothetical preference relations*. More precisely, for some member ℓ, we denote by H_0^ℓ the hypothetical preference of that member. The expression "$q_s \, H_0^\ell \, q_t$" means that we hypothesize that member ℓ *directly prefers* the bundle q_s over the bundle q_t (for $s, t \in N$). In Cherchye et al. (2007) these hypothetical relations are then used to derive necessary conditions that the data need to satisfy in order to be compatible with the general collective consumption model. We refer to Cherchye et al. (2007, 2012) for a detailed discussion.

Given our specific objective, we consider an extension of WARP to 2-member households that makes use of this notion of hypothetical preferences. This extension is derived from the revealed preference characterization in Proposition 2 of Cherchye et al. (2007), by essentially dropping the transitivity requirement.

Definition 6.5.1. GENERAL 2-WARP
Let $S = \{(p_t, q_t) : \ t \in N\}$ be a dataset of a two-member household. We say that S is consistent with general 2-WARP *if and only if there exist hypothetical preferences H_0^1, H_0^2 that satisfy:*

(a) *For each pair of distinct observations $s, t \in N$: if $p_s q_s \geq p_s q_t$, then $q_s \, H_0^1 \, q_t$ or $q_s \, H_0^2 \, q_t$;*

(b) *For each pair of distinct observations $s, t \in N$: if $p_s q_s \geq p_s q_t$, $q_t \, H_0^\ell \, q_s$ and $q_t \neq q_s$, then $q_s \, H_0^r \, q_t$, with $\ell, r \in \{1, 2\}$ and $\ell \neq r$;*

(c) *For each three distinct observations $s, t, u \in N$: if $p_s q_s \geq p_s \, (q_t + q_u)$*

and $q_t \, H_0^{\ell} \, q_s$*, then* $q_s \, H_0^{r} \, q_u$ *with* $\ell, r \in \{1, 2\}$ *and* $\ell \neq r$*;*

(d) *For each pair of distinct observations* $s, t \in N$*: if* $q_s H_0^1 q_t$*,* $q_s H_0^2 q_t$ *and* $q_t \neq q_s$*, then* $p_t q_t < p_t q_s$*;*

(e) *For each three distinct observations* $s, t, u \in N$*: if* $q_s H_0^1 q_t$ *and* $q_u H_0^2 q_t$*, then* $p_t q_t < p_t (q_s + q_u)$*.*

In what follows, we will show that it is possible to check this general 2-WARP condition efficiently (i.e. in polynomial time), which contrasts with our results for the private and public settings in the previous sections. Importantly, we will only show this complexity result for the two-member case. Differing from before, the result is not straightforwardly generalized towards the general case with L household members ($L \geq 2$). We leave the study of this L-member case for future research.[4]

As a first step towards formulating the decision problem corresponding to our definition of general 2-WARP, we define a simplification of the above definition that is easier to use. Specifically, we replace condition (b) with a closely similar, but somewhat more stringent condition, and we drop conditions (d) and (e).

Definition 6.5.2. GENERAL 2-WARP
Let $S = \{(p_t, q_t) : \ t \in N\}$ *be a dataset of a two-member household. We say that* S *is consistent with* general 2-WARP *if and only if there exist hypothetical preferences* H_0^1, H_0^2 *that satisfy:*

(i) *For each pair of distinct observations* $s, t \in N$*: if* $p_s q_s \geq p_s q_t$*, then* $q_s \, H_0^1 \, q_t$ *or* $q_s \, H_0^2 \, q_t$*;*

(ii) *For each pair of distinct observations* $s, t \in N$*: if* $p_s q_s \geq p_s q_t$*,* $q_t \, H_0^{\ell} \, q_s$ *and* $q_t \neq q_s$*, then* $\neg(q_s \, H_0^{\ell} \, q_t)$*, with* $\ell \in \{1, 2\}$*;*

[4] In this respect, we note that existing applications of the collective model usually consider households with only two decision makers (e.g. husband and wife, with expenses on children treated as public consumption). Therefore, we may safely argue that the two-member case is the most relevant one from a practical perspective.

(iii) *For each three distinct observations $s, t, u \in N$: if $p_s q_s \geq p_s (q_t + q_u)$ and $q_t \, H_0^\ell \, q_s$, then $q_s \, H_0^r \, q_u$ with $\ell, r \in \{1, 2\}$ and $\ell \neq r$;*

Claim 6.5.1. *Definitions 6.5.1 and 6.5.2 are equivalent.*

Proof. $\Rightarrow$) Assume that there exists hypothetical relations for which conditions (a)-(e) in Definition 6.5.1 are satisfied. It is clear that conditions (i) and (iii) are then also satisfied, as these are identical to (a) and (c). Now suppose the hypothetical relations include a violation of (ii), i.e., there exist distinct observations $s, t \in N$, for which $p_s q_s \geq p_s q_t$, $q_t \, H_0^1 \, q_s$, $q_s \, H_0^1 \, q_t$ and $q_t \neq q_s$. We then have to consider two scenarios: either $p_t q_t \geq p_t q_s$ or $p_t q_t < p_t q_s$. If $p_t q_t < p_t q_s$, then there is no need to specify $q_t \, H_0^1 \, q_s$ and thus condition (ii) is by construction satisfied. In the alternative scenario, $p_t q_t \geq p_t q_s$, then $q_s \, H_0^1 \, q_t$ implies that $q_t \, H_0^2 \, q_s$ (since condition (b) is satisfied). But this entails a violation of condition (d), since $p_s q_s \geq p_s q_t$, $q_t \, H_0^1 \, q_s$ and $q_t \, H_0^2 \, q_s$. This gives us the desired contradiction.

$\Leftarrow$) Next assume there exist hypothetical relations for which Conditions (i)-(iii) in Definition 6.5.2 are satisfied. Again, conditions (a) and (c) are identical to (i) and (iii) and are satisfied. Next, if (b) is violated, there exist $s, t \in N$ such that $p_s q_s \geq p_s q_t$, $q_t \, H_0^1 \, q_s$, $q_t \neq q_s$, and $\neg(q_s \, H_0^2 \, q_t)$. Since condition (i) requires either $q_s \, H_0^1 \, q_t$ or $q_s \, H_0^2 \, q_t$, it must be the case that $q_s \, H_0^1 \, q_t$, which violates (ii), thus there is violation of (b) if (i)-(iii) is satisfied. Now suppose (d) is violated. Then $q_s \, H_0^1 \, q_t$, $q_s \, H_0^2 \, q_t$ and $p_t q_t \geq p_t q_s$. By rule (i), it must then be the case that $q_t \, H_0^1 \, q_q$ or $q_t \, H_0^2 \, q_q$, either of which again violates (ii). Finally, a violation of (e) implies $q_s H_0^1 q_t$, $q_u H_0^2 q_t$ and $p_t q_t \geq p_t (q_s + q_u)$. To satisfy condition (iii), if $p_t q_t \geq p_t (q_s + q_u)$ and $q_s H_0^1 q_t$, it must be the case that $q_t H_0^2 q_u$, since we also have $q_u H_0^2 q_t$, either (ii) or (iii) must be violated if (e) is violated. □

The problem of testing whether a collective rationalization of S exists is then formulated as the following *decision problem*:

Problem 6.5.1. GENERAL 2-WARP

Instance: A dataset $S := \{(p_t, q_t) : t \in N\}$.
Question: Do there exist hypothetical preferences H_0^1, H_0^2*, such that conditions (i) - (iii) in Definition 6.5.2 hold?*

Before studying this decision problem more in detail, we want to make the following remarks. If the dataset S contains only three observations, let us say s, t, and u, then the answer to the decision problem is No if and only if the following three inequalities hold: $p_s q_s \geq p_s(q_t + q_u)$, $p_t q_t \geq p_t(q_s + q_u)$, and $p_u q_u \geq p_u(q_s + q_t)$. For datasets containing more than three observations, however, the presence of these three inequalities is not necessary to have a No answer. Indeed, the reader can check that the following inequalities involving four observations, let us say s, t, u, and v, also leads to a No answer to 2-WARP: $p_s q_s \geq p_s q_t$, $p_t q_t \geq p_t(q_s + q_u)$, $p_t q_t \geq p_t(q_s + q_v)$, $p_u q_u \geq p_u(q_t + q_v)$, and $p_v q_v \geq p_v(q_t + q_u)$. Furthermore, we mention that if there is no inequality of the form $p_s q_s \geq p_s(q_t + q_u)$ for all triples s, t, and u in N then we have a YES instance of 2-WARP.

6.5.1 A graph interpretation of 2-WARP

We translate conditions (i) to (iii) into a directed graph setting (see Talla Nobibon et al. (2011) for a related construction). We build a directed graph $G = (V, A)$ from the dataset $S := \{(p_t, q_t) : t \in N\}$ as follows. A pair of distinct observations (s, t) with $s, t \in N$ represents a vertex in V if and only if both $p_s q_s \geq p_s q_t$ and $p_t q_t \geq p_t q_s$. Notice that V contains $O(n^2)$ vertices and if the vertex (s, t) exists then the vertex (t, s) also exists. The set of arcs A is defined in two steps as follows:

1: First, there is an arc from a vertex (s, t) to a vertex (u, v) whenever $t = u$.

2: Second, for any three distinct observations $s, t, u \in N$ satisfying $p_s q_s \geq p_s(q_t + q_u)$, $p_t q_t \geq p_t q_s$, $p_u q_u \geq p_u q_s$, we have an arc from (s, u) to (t, s), and from (s, t) to (u, s).

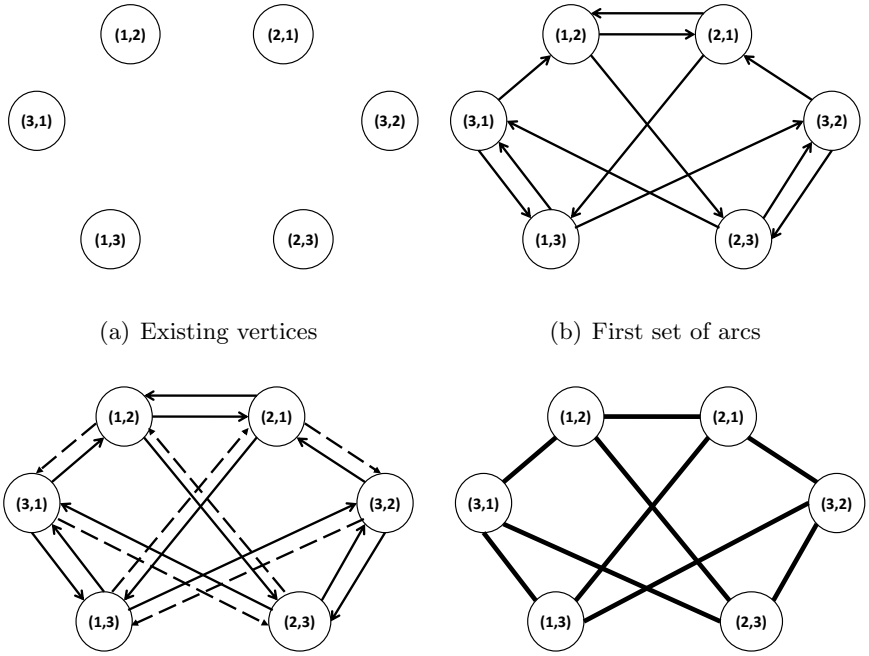
Notice that Step 1 ensures that there is an arc from node (s,t) to node (t,s) and vice versa. This graph construction differs from the one used when checking whether a dataset of a unitary household satisfies WARP: in that case, a directed graph is built where a vertex corresponds with an observation and there is an arc from s to t if and only if $p_s q_s \geq p_s q_t$. That approach is not considered because it is not quite clear how to deal with inequalities of the form $p_s q_s \geq p_s(q_t + q_u)$.

Given the directed graph $G = (V, A)$ built above, we define the 2-*undirected graph* $G_2 = (V, E)$ associated with G as the undirected graph obtained from G by transforming any pair of arcs forming a cycle of length 2 into a single edge (undirected arc); more precisely, $\{v_1, v_2\} \in E$ if and only if $v_1 v_2 \in A$ and $v_2 v_1 \in A$.

As an illustration of the graph construction, consider a dataset with three observations satisfying: $p_1 q_1 \geq p_1(q_2 + q_3)$, $p_2 q_2 \geq p_2(q_1 + q_3)$ and $p_3 q_3 \geq p_3(q_1 + q_2)$. This implies the existence of the vertices depicted in Figure 6.1(a). The arcs stemming from Step 1 appear in Figure 6.1(b), and the final graph is depicted in Figure 6.1(c), where the dashed arcs are derived from Step 2. Finally, the 2-undirected graph G_2 associated with G is depicted in Figure 6.1(d). We have the following result.

Theorem 6.5.1. *S is a* YES *instance of* 2-WARP *if and only if the* 2*-undirected graph G_2 associated with G is bipartite.*

Proof. $\Leftarrow$) Suppose that G_2 is bipartite. Thus, the set of vertices V can be partitioned into two subsets V_1 and V_2 such that each subset induces an independent set. In other words, $V = V_1 \cup V_2$, $V_1 \cap V_2 = \varnothing$, and there is no edge between two vertices of V_1 and no edge between two vertices of V_2. We build the hypothetical preferences H_0^1 and H_0^2 as follows: for every vertex $(s,t) \in V_1$ (respectively $(s,t) \in V_2$) we have $q_s \, H_0^1 \, q_t$ $\big($respectively $q_s \, H_0^2 \, q_t\big)$. Furthermore, for two distinct observations s and t such that $p_s q_s \geq p_s q_t$ and $(s,t) \notin V$, we set $q_s \, H_0^1 \, q_t$ and $q_s \, H_0^2 \, q_t$. This completes the definition of H_0^1 and H_0^2. Notice that there is no distinct pair of observations s, t for which we set $q_s H_0^\ell q_t$ and $q_t H_0^\ell q_s$ for some $\ell \in \{1, 2\}$. We now argue that



(a) Existing vertices

(b) First set of arcs

(c) Final graph G

(d) 2-undirected graph G_2

Figure 6.1: *Illustration of the construction of G and the associated 2-undirected graph G_2*

H_0^1 and H_0^2 satisfy conditions (i) to (iii).

Condition (i): Let $s, t \in N$ be two distinct observations such that $p_s q_s \geq p_s q_t$. On the one hand, if $(s,t) \notin V$ then, by construction, $q_s \, H_0^1 \, q_t$ and $q_s \, H_0^2 \, q_t$. On the other hand, if $(s,t) \in V = V_1 \cup V_2$ then $(s,t) \in V_1$ or $(s,t) \in V_2$, and hence $q_s \, H_0^1 \, q_t$ or $q_s \, H_0^2 \, q_t$. Thus condition (i) is satisfied.

Condition (ii): As described above, there is no distinct pair of observations $s, t \in N$ for which we set $q_s H_0^\ell q_t$ and $q_t H_0^\ell q_s$ for some $\ell \in \{1, 2\}$. Thus condition (ii) is satisfied.

Condition (iii): Let $s, t, u \in N$ be three distinct observations such that $p_s q_s \geq p_s(q_t + q_u)$ and $q_t \, H_0^1 \, q_s$. There are two cases: (1) if $p_u q_u < p_u q_s$ then $(s, u) \notin V$ and, since $p_s q_s \geq p_s q_u$, we have by construction of H_0^ℓ, $q_s \, H_0^1 \, q_u$ and $q_s \, H_0^2 \, q_u$, and we are done; (2) if $p_u q_u \geq p_u q_s$ then $(s, u) \in V$. Let us now argue by contradiction that $(t, s) \in V$. Indeed, if $(t, s) \notin V$, then $(s, t) \notin V$. That however, is impossible since $p_s q_s \geq p_s q_t$, and we would have had by construction $q_s \, H_0^1 \, q_t$ and $q_s \, H_0^2 \, q_t$, which cannot be reconciled with $q_t \, H_0^1 \, q_s$. Thus $(t, s) \in V$, and in fact, since $q_t \, H_0^1 \, q_s$, $(t, s) \in V_1$. Following the construction of G, we have an arc from (t, s) to (s, u) and an arc from (s, u) to (t, s) (because $p_s q_s \geq p_s(q_t + q_u), p_t q_t \geq p_t q_s, p_u q_u \geq p_u q_s$). Therefore, there is an edge between the vertices (t, s) and (s, u) in G_2, and we conclude that $(s, u) \in V_2$, which implies that $q_s \, H_0^2 \, q_u$. This completes the verification of condition (iii).

$\Rightarrow$) Now, we suppose that S is a YES instance of 2-WARP; there exist H_0^1 and H_0^2 satisfying conditions (i) to (iii). We want to show that the 2-undirected graph G_2 is bipartite. In other words, we want to partition V into two subsets V_1 and V_2 such that there is no edge between two vertices of V_1 and no edge between two vertices of V_2.

Given H_0^1 and H_0^2 we set the vertices in V_1 (respectively in V_2) as follows: a vertex $(s, t) \in V$ belongs to V_1 (respectively to V_2) if $q_s \, H_0^1 \, q_t$ $\big($respectively $q_s \, H_0^2 \, q_t\big)$. It is not difficult to see that $V_1 \cap V_2 = \varnothing$ and that any vertex in V is either in V_1 or in V_2. Hence, V_1 and V_2 constitute a valid partition of V. We argue, by contradiction, that V_1 and V_2 induce independent sets. Without loss of generality, suppose V_1 is not an independent set. There exist two vertices (s, t) and (u, v) in V_1 with an edge between them in G_2. Thus, in the graph G there is an arc from (s, t) to (u, v), and from (u, v) to (s, t). If both arcs originate from Step 1, we have $u = t$ and $v = s$, which implies $(s, t) \in V_1$, and $(t, s) \in V_1$ which can only happen if $q_s \, H_0^1 \, q_t$ and $q_t \, H_0^1 \, q_s$; this, however, contradicts condition (ii) for H_0^1. If both arcs originate from Step 2, we also have $u = t$ and $v = s$, and the same argument applies. Hence, one arc originates from Step 1 and one arc originates from

Step 2. Without loss of generality, we can assume that the arc from (s,t) to (u,v) comes from Step 1, while the arc from (u,v) to (s,t) comes from Step 2. This implies that $u = t$, and apparently $p_t q_t \geq p_t(q_s + q_v)$. Since $q_s \, H_0^1 \, q_t$, condition (iii) implies that $q_t \, H_0^2 \, q_v$. By hypothesis, we have $q_t \, H_0^1 \, q_v$ and $p_v q_v \geq p_v q_t$ (because $(t,v) \in V_1$). From condition (i) we know that $q_v \, H_0^1 \, q_t$ or $q_v \, H_0^2 \, q_t$. This, together with $q_t \, H_0^1 \, q_v$ and $q_t \, H_0^2 \, q_v$, implies that either $q_t \, H_0^1 \, q_v$ and $q_v \, H_0^1 \, q_t$ or $q_t \, H_0^2 \, q_v$ and $q_v \, H_0^2 \, q_t$. In the first case, H_0^1 violates condition (ii) whereas in the second case H_0^2 violates condition (ii). In both cases, we have a contradiction with condition (ii). This concludes the proof of Theorem 6.5.1.

□

6.5.2 An Algorithm for 2-WARP

We present an algorithm for 2-WARP that is based on Theorem 6.5.1. The pseudocode is described by Algorithm 6.

Algorithm 6 *Algorithm for* 2-WARP

1: build the directed graph G from the dataset S
2: build the 2-undirected graph G_2 associated with G
3: **if** G_2 is bipartite **then** return YES, **else** return NO

It is clear that each of the three steps of Algorithm 6 can be done in polynomial time. Thus, we have the following result:

Theorem 6.5.2. *Algorithm 6 solves* 2-WARP *in polynomial time.*

6.6 Conclusion

We studied three alternative extensions of the weak axiom of revealed preference (WARP) that apply to the collective consumption model. We proved that for the private and public settings, the corresponding testing problem is NP-COMPLETE even for two (but also for more) household members.

However, for the general setting, testing 2-WARP can be done in polynomial time for households consisting of two members. When there are three or more household members, the complexity of the testing problem for this general setting remains an open question.

Chapter 7

Testing Stochastic Models of Preference

7.1 Introduction

A major challenge for testing theories of choice behaviour comes from the fact that decision makers often show inconsistent behaviour when faced with repetitions of the same choice situations. Even when faced with relatively simple choices, such as binary choices between two alternatives, and short time spans between repetitions, it is fairly common to observe a decision maker switching between choice alternatives. Such inconsistencies are often explained in two different ways. The first way is that even though decision makers have deterministic preferences over alternatives, the choices they actually make are probabilistic. Under this model the decision maker is thought to sometimes make errors, choosing a less preferred alternative by mistake. A second theory is that the decision maker does choose according to his preferences, but that these preferences themselves are probabilistic. In other words, the decision maker does not make errors, she just changes her mind according to some underlying probability func-

This chapter is the result of a collaboration with Clintin Davis-Stober, Michel Regenwetter and Frits C.R. Spieksma.

tion. In this paper, we will be working with the second of these models, which we will refer to as *mixture models* (Regenwetter et al., 2010, 2011).

Compared to previous chapters, there will be some differences in the setting. There is a finite set of choice alternatives over which the decision maker has preferences, rather than a continuum of bundles over which there is a utility function. Every observed decision situation will also be binary, the decision maker is given only two alternatives from the set and will be asked to choose one or the other. In this chapter, we will allow any strict linear order over the alternatives, i.e., there are no restrictions on the preferences a decision maker can have. The model we will test is thus a mixture model of strict linear preferences, for ease of reading, we will just refer to it as a mixture model. An important result for testing this model is given by Suck (1992), he shows that testing this mixture model is equivalent to testing membership of the *linear ordering polytope*. It is shown by Grötschel et al. (1981, 1993) that a polynomial time algorithm for membership implies a polynomial time algorithm for optimization over the polytope. Since optimization over this polytope corresponds to the well known NP-HARD LINEAR ORDERING problem, the result by Suck implies that testing whether data are consistent with the mixture model is NP-HARD.

Our contribution in this chapter is a column generation based algorithm to test the mixture model. Such an algorithm is interesting, not only for the problem at hand, but also in general, as it can be easily adapted for tests of different mixture models with other forms of preferences. The rest of this chapter will unfold as follows. In Section 7.2, we lay out the notation and definitions used and we present the setting and model in more detail. Section 7.3 will consist of a basic description of the column generation algorithm. The implementation and results of this algorithm will be discussed in Section 7.4. Section 7.5 handles Bayes factor calculation. The Bayes factor is a measure for the likelihood that observed choices are the result of an underlying stochastic choice process. Finally, Section 7.6 concludes.

7.2 Notation and Definitions

Consider a set A, consisting of n alternatives. For each ordered pair of distinct alternatives $(i, j) \in A^2$, we consider a non-negative number $p_{ij} \leq 1$. These numbers represent the probability that i is chosen over j. As we assume strict preferences, $p_{ij} + p_{ji} = 1$ for each pair of distinct $i, j \in A$. We refer to the set of numbers p_{ij} as the *data*. Preference orderings over the alternatives are represented by the relation $\succ$, with $i \succ j$ if i is preferred over j. These relations are strict, complete and transitive. We will use the index m to denote a particular preference ordering. The set of all preference orderings is O. We further consider the subsets O_{ij}, for which $m \in O_{ij}$ if $i \succ_m j$. A mixture model of preference assumes that when a decision maker is faced with a choice, each preference ordering has a certain probability of being used to make the choice. When these probabilities are consistent with the numbers p_{ij}, we say that the model rationalizes the observed data.

Definition 7.2.1. *Data can be rationalized by a mixture model of strict linear preferences if and only if there exists a probability* $0 \leq x_m \leq 1, \forall m \in O$ *for which:*

$$\sum_{m \in O_{ij}} x_m = p_{ij}, \qquad \forall (i, j) \in A^2, i \neq j \tag{7.1}$$

A straightforward test of the mixture model is thus to check whether a solution exists to the system of equalities (7.1). However, this system of equalities has a variable for every possible linear order over the alternatives, of which there exist $n!$. For even moderate numbers of alternatives, it is computationally intensive to solve this system. Suck's result that testing this mixture model is equivalent to a membership problem of the linear ordering polytope, suggests a second approach. A polytope can also be described by facet defining inequalities. This means that, if and only if the p_{ij} values satisfy all facet defining inequalities of the linear ordering polytope, the mixture model is satisfied. However, the problem of finding a violated

facet defining inequality is also NP-HARD for several known classes of these inequalities. Furthermore, the number of classes rises exponentially with the number of alternatives. The largest number of alternatives for which all classes of facet defining inequalities of the linear ordering polytope are known is 7 (Martí and Reinelt, 2011). The goal of this chapter is to present a test of the mixture model that is capable of handling datasets for which both testing the system of equalities (7.1) or checking the facet defining inequalities is possible, which we will do in the next section.

7.3 Column Generation

In this section, we will describe an algorithm based on column generation to test the mixture model. The advantage of using column generation, is that it avoids having to consider all of the variables at once. Given the exponential number of these variables, this will allow us to handle datasets of sizes for which solving (7.1) directly is impractical. To be able to use a column generation approach, we will first reformulate the system (7.1) into a linear programming problem. For this linear program, we can then find the dual formulation and, using both of these problems, devise a column generation algorithm.

7.3.1 Linear Programming Formulation

In the previous section, we provided a system of equalities, which are both necessary and sufficient for the data to be rationalizable by a model of stochastic strict linear ordering preferences. This system can be easily rewritten as a linear programming problem as follows.

$$\text{Minimize} \qquad z \tag{7.2}$$

Subject to

$$\sum_{m \in O_{ij}} x_m + z \geq p_{ij} \qquad \forall (i,j) \in A^2, i \neq j \tag{7.3}$$

$$\sum_{m \in O} x_m \leq 1 \tag{7.4}$$

$$x_m, z \geq 0 \qquad \forall m \in O \tag{7.5}$$

Claim 7.3.1. *The optimal solution value of (7.2)-(7.5) is equal to 0 if and only if a solution exists to the system of equalities (7.1).*

Proof. The claim can be easily checked as follows. First, we show that if there exists a solution with $z = 0$, the values of the x_m variables also form a solution to (7.1). Finally, we show that any solution to (7.1) is also a solution to (7.2)-(7.5), with $z = 0$.

Suppose we have a solution to (7.2)-(7.5) with solution value 0. As $1 \geq \sum_{m \in O} x_m = \sum_{m \in O_{ij}} x_m + \sum_{m \in O_{ji}} x_m = 1$, we must have $\sum_{m \in O_{ij}} x_m = p_{ij}$ and $\sum_{m \in O_{ji}} x_m = p_{ji}$. The values x_m are thus a solution to (7.1). Finally, suppose we have a solution to the system of equalities (7.1). The values of x_m can then be put into the linear programming problem. As for each ordered pair (i, j), we have $\sum_{m \in O_{ij}} x_m = p_{ij}$ and $\sum_{m \in O} x_m = 1$, it is clear that the constraints (7.3) are met with $z = 0$. □

Clearly this LP-formulation still has the same large number of variables ($n!$) as (7.1). However, this formulation has a relatively small number of constraints (n^2). Since an optimal solution to an LP can be found with a number of non-zero variables less than or equal to the number of constraints, it is clear that not all variables are needed (Chvátal, 1983). We will therefore use a column generation approach. In this context, we will refer to the linear problem (7.2)-(7.5) as the primal or master problem.

7.3.2 The Pricing Problem

For each linear programming problem, there exists an associated *dual* problem. The dual problem associated with (7.2)-(7.5) is as follows.

$$\text{Maximize} \quad \sum_{(i,j)\in A^2, i\neq j} p_{ij} y_{ij} - c \tag{7.6}$$

Subject o

$$\sum_{(i,j):m\in O_{ij}} y_{ij} - c \leq 0 \qquad \forall m \in O \tag{7.7}$$

$$\sum_{(i,j)\in A^2, i\neq j} y_{ij} \leq 1 \tag{7.8}$$

$$y_{ij}, c \geq 0 \qquad \forall (i,j) \in A^2, i \neq j \tag{7.9}$$

An informal description of a column generation approach tailored to solving (7.2)-(7.5) is as follows. Suppose we have a feasible solution (x, z) to the primal problem as well as an associated dual solution (y, c). It is well-known that feasibility of this dual solution (y, c) is equivalent to optimality of the primal solution (x, z). In other words, if (x, z) is the optimal solution to the master problem, then the associated dual solution (y, c) will be feasible and vice versa. Thus, if we want to test optimality of (x, z), we may test whether (y, c) satisfies (7.7), (7.8) and (7.9). In a column generation approach, we consider a restricted master, that is, many of the primal variables from (7.2)-(7.5) are not considered explicitly. Solving this restricted master gives a feasible primal solution (x, z), as well as a correspoding dual solution (y, c). If this dual solution is feasible, then this is proof that the solution to the restricted master problem is also an optimal solution to the master problem. Otherwise, a violated dual constraint is identified which gives rise to a primal variable that should be included in the restricted master.

Checking whether the solution (y, c) is a feasible solution to the dual problem is done by solving a pricing problem. In this case, there exists a violated inequality if and only if there exists a linear order $m \in O$, for

which $\sum_{ij:m\in O_{ij}} y_{ij} > c$. This gives us the following pricing problem. The variables b_{ij} represent preference relations for an ordered pair of distinct alternatives, $(i,j) \in A^2$: if $b_{ij} = 1$, then i is preferred over j.

$$\text{Maximize} \quad \sum_{(i,j)\in A^2, i\neq j} y_{ij} b_{ij} \tag{7.10}$$

Subject to

$$b_{ij} + b_{ji} = 1 \qquad \forall (i,j) \in A^2, i \neq j \tag{7.11}$$

$$b_{ij} + b_{jk} + b_{ki} \leq 2 \qquad \forall (i,j,k) \in A^3, i \neq j \neq k \neq i \tag{7.12}$$

$$b_{ij} \in \{0,1\} \qquad \forall (i,j) \in A^2, i \neq j \tag{7.13}$$

A solution of the problem consists of the b_{ij} variables, which given the constraints (7.11) - (7.13) encode a strict linear order. Any such solution for which the objective value (7.10) is greater than c, corresponds to a violated inequality (7.7) of the dual. This violated inequality directly corresponds to a primal variable, which when added to the restricted master problem will improve its solution. We notice that this pricing problem is the well known linear ordering problem (see Martí and Reinelt (2011)). This problem is known to be NP-Hard (Garey and Johnson, 1979).

7.3.3 Column Generation Algorithm

We are now in a position to describe the full column generation algorithm, as shown in Algorithm 7. The general idea is as follows. Initially, we solve a restricted master problem. This problem only contains a subset of all variables of the full master. Given a solution to the restricted master, we test whether this solution is optimal for the full master problem, by checking whether there exist violated constraints in the dual problem. This is done by solving the pricing problem. If the pricing problem identifies no violated constraints, this means the optimal solution to the master problem has been reached. If a violated constraint is identified, the associated

variable is added to the restricted master, which is then resolved. Since it is clear that there can be no solution to the master problem with $z < 0$, the column generation algorithm can also terminate as soon as $z = 0$.

Algorithm 7 Column Generation Algorithm

```
1: Solve Restricted Master Problem.
2: if z = 0 then
3:    Optimal solution found: stop
4: else
5:    Update Pricing Problem with values y_ij.
6:    Solve Pricing Problem
7:    if Value Pricing Solution ≤ c then
8:       Optimal solution found: stop
9:    else
10:      Add variable corresponding to the linear order found to the Re-
         stricted Master Problem
11:      GOTO Line 1
12:   end if
13: end if
```

7.3.4 Stopping Condition

To finish this section on Column Generation, we note that we are not interested in finding an optimal solution to the master problem. We are only interested in the decision problem, whether or not a feasible solution to the master problem exists in which $z = 0$ (and as a result, every constraint is met with equality). This subtle difference allows us to use an extra stopping condition, based on the objective value of the pricing solutions.

Theorem 7.3.1. *Given numbers y_{ij} for all pairs $(i, j) \in A^2$ $(i \neq j)$, there is a solution to the system (7.1) only if there exists a linear order $m \in O$,*

such that:

$$\sum_{(i,j)\in A^2 : i \succ_m j} y_{ij} \geq \sum_{(i,j)\in A^2, i\neq j} y_{ij} p_{ij} \tag{7.14}$$

Proof. We argue by contradiction. Suppose that there exists a feasible solution (7.1), while for each $m \in O$:

$$\sum_{(i,j)\in A^2 : i \succ_m j} y_{ij} < \sum_{(i,j)\in A^2, i\neq j} y_{ij} p_{ij}. \tag{7.15}$$

Thus, since $x_m \geq 0$ and $\sum_{m\in O} x_m = 1$:

$$\sum_{m\in O} (x_m \sum_{(i,j)\in A^2 : i \succ_m j} y_{ij}) < \sum_{(i,j)\in A^2, i\neq j} y_{ij} p_{ij}. \tag{7.16}$$

The left hand side is equal to:

$$\sum_{m\in O} (x_m \sum_{(i,j)\in A^2 : i \succ_m j} y_{ij}) = \sum_{(i,j)\in A^2} \sum_{m\in O_{ij}} x_m y_{ij}. \tag{7.17}$$

Thus we have derived the validity of the following inequality:

$$\sum_{(i,j)\in A^2, i\neq j} \sum_{m\in O_{ij}} y_{ij} x_m < \sum_{(i,j)\in A^2, i\neq j} y_{ij} p_{ij} \tag{7.18}$$

Now, since (7.1) is satisfied, we have for each ordered pair $(i, j) \in A^2, i \neq j$ that:

$$\sum_{m\in O_{ij}} x_m = p_{ij}. \tag{7.19}$$

Multiplying both sides by y_{ij} preserves the equality, thus for each ordered pair $(i, j) \in A^2, i \neq j$ we have

$$\sum_{m\in O_{ij}} y_{ij} x_m = y_{ij} p_{ij}. \tag{7.20}$$

And summing over all ordered pairs gives

$$\sum_{(i,j)\in A^2, i\neq j} \sum_{m\in O_{ij}} y_{ij} x_m = \sum_{(i,j)\in A^2, i\neq j} y_{ij} p_{ij}, \tag{7.21}$$

which contradicts (7.18) □

This result allows us to end the column generation algorithm if there is no linear order $m \in O$ for which $\sum_{(i,j)\in A^2:i\succ_m j} y_{ij} > \sum_{(i,j)\in A^2,i\neq j} y_{ij}p_{ij}$. It can easily be seen that this stopping condition is stronger than standard column generation stopping condition, which is ending the algorithm if there is no linear order $m \in O$ for which $\sum_{(i,j)\in A^2:i\succ_m j} y_{ij} > c$. Indeed, each dual solution (y, c) will satisfy $\sum_{(i,j)\in A^2,i\neq j} p_{ij}y_{ij} > c$, since the value of the objective function of the restricted master (z) is equal to the value of the objective function of the dual problem $(\sum_{(i,j)\in A^2,i\neq j} p_{ij}y_{ij} - c)$. As long as $z > 0$, it is thus the case that $\sum_{(i,j)\in A^2,i\neq j} p_{ij}y_{ij} > c$.

We note that this result can be seen as an application of Farkas' Lemma. Indeed, consider this slight rephrasing of the original system of equalities, in which we make the convexity constraint explicit.

$$\sum_{m\in O_{ij}} x_m = p_{ij} \qquad \forall (i,j) \in A^2, i \neq j \tag{7.22}$$

$$\sum_{m\in O} x_m = 1 \tag{7.23}$$

$$x_m \geq 0 \qquad \forall m \in O \tag{7.24}$$

Then Farkas' Lemma states this system has a solution if and only if there does not exist a solution (f_{ij}, g) to the following system of inequalities.

$$\sum_{i,j\in O:i\succ_m j} f_{ij} + g < 0 \qquad \forall m \in O \tag{7.25}$$

$$\sum_{(i,j)\in A^2,i\neq j} p_{ij}f_{ij} + g \geq 0 \tag{7.26}$$

However, given a dual solution (y_{ij}, c) for which condition (7.14) does not hold, i.e., for which no linear order $m \in O$ exists satisfying (7.14), it is clear that a solution exists to (7.25-7.26). Indeed, let $f_{ij} = y_{ij}$, then for each $m \in O$, it is the case that

$$\sum_{i,j\in O:i\succ_m j} f_{ij} < \sum_{(i,j)\in A^2,i\neq j} p_{ij}f_{ij}. \tag{7.27}$$

Given this, there must exist some g, such that

$$\sum_{i,j \in O: i \succ_m j} f_{ij} + g < 0 \leq \sum_{(i,j) \in A^2, i \neq j} p_{ij} f_{ij} + g. \tag{7.28}$$

Furthermore, if condition (7.14) does not hold, Theorem 7.3.1 in effect states that there is a lower bound $z' > 0$ on the optimal solution z^* of the master problem. In general, the convexity constraint on the x_m variables allows us to derive a lower bound on the master problem in every iteration of the column generation algorithm as follows (See Bazaraa et al. (2011), Lübbecke and Desrosiers (2005)). Let $\bar{z}$ be the objective value of an optimal solution to the restricted master problem, and b_{ij} the optimal solution to the pricing problem. Then it is the case that

$$\bar{z} - (\sum_{(i,j) \in A^2} (b_{ij} y_{ij}) - c) \leq z^*. \tag{7.29}$$

Since

$$\bar{z} = \sum_{(i,j) \in A^2, i \neq j} (p_{ij} y_{ij}) - c, \tag{7.30}$$

we have

$$\sum_{(i,j) \in A^2, i \neq j} (p_{ij} y_{ij}) - \sum_{(i,j) \in A^2, i \neq j} (b_{ij} y_{ij}) \leq z^*. \tag{7.31}$$

7.4 Implementation

In this section we discuss the implementation of the column generation algorithm. Obviously, we can run Algorithm 7 by making use of standard linear and integer program solvers. However, as suboptimal solutions to the pricing problem may also identify violated inequalities of the dual, we also use fast heuristic approaches to solve the pricing problem. We describe these heuristic algorithms in subsection 7.4.1. Subsection 7.4.2 then contains descriptions of our datasets. Finally, subsection 7.4.3 gives results on computation times for the various algorithms and datasets.

7.4.1 Heuristic Algorithms

As the linear ordering problem is a well-known and well-studied NP-HARD problem, an extensive literature exists on heuristics. In this section, we describe our implementation based on best insertion constructive and insertion local search algorithms (Laguna et al., 1999). In the literature, it has been observed that these perform well when compared to other simple heuristics (See Martí and Reinelt (2011)). We use multi-start procedures, varying the order in which alternatives are added in the insertion heuristics. In this way, we have multiple solutions to compare against each other. These multiple solutions can be used to either pick the best solution, or to identify multiple variables to add to the restricted master in the column generation. Furthermore, we describe an algorithm for adjusting solutions to the pricing problem, such that the resulting linear orders more closely resemble the observed preferences.

Best Insertion Heuristics

We describe a constructive heuristic called Best Insertion (Algorithm 8); and a local search method based on a 'move' neighbourhood. The descriptions of these algorithms are based on Martí and Reinelt (2011) The Best Insertion algorithm creates an initial ranking of the alternatives, by iteratively placing alternatives in an ordering over a (sub)set of the alternatives. In the local search method, the position of alternatives in this ordering can be changed by local moves. Initially, we consider a set of all alternatives A. For every ordered pair of alternatives, $(i, j) \in A^2$, the value of placing i before j is given by y_{ij}. A linear order is denoted by $\langle a, b, \ldots \rangle$.

In our implementation of Algorithm 8, the choice of alternatives $i \in A$ in line 7 is made as follows. If, initially, an alternative $j \in A$ is chosen in step 2, we pick alternatives $j+1, \ldots, n, 1, \ldots, j-1$, in that order, in step 7.

For the local search algorithm, let $v(a)$ be the objective value associated

Algorithm 8 Best Insertion Heuristic

INPUT: Set A and numbers y_{ij} for each ordered pair $(i, j) \in A^2, i \neq j$.
Select an alternative $i \in A$ and set $A := A \backslash \{i\}$.
Create an ordering $\langle i \rangle$.
Set $k := 1$
while $A \neq \varnothing$ **do**
  Let $\langle a_1, a_2, \ldots, a_k \rangle$ denote the current ordering.
  Choose an alternative $i \in A$.
  For each $t = 1, \ldots, k+1$, compute $q_t = \sum_{j=1}^{t-1} y_{a_j,i} + \sum_{j=t}^{k} y_{i,a_j}$.
  Let $r = \arg\max_{t=1,\ldots,k+1} q_t$.
  Set $j := k + 1$
  while $j > r + 1$ **do**
    $a_j := a_{j-1}$
    $j := j - 1$
  end while
  $a_r := i$
  $A := A \backslash \{i\}$
end while
OUTPUT: A linear order $a = \langle a_1, a_2, \ldots, a_n \rangle$

with a linear order $\langle a_1, a_2, \ldots, a_n \rangle$. Furthermore, let $v(a, i, j)$, be the value of the order that results when the object in position i is moved to position j. Algorithm 9 gives the full pseudo-code for the local search.

Algorithm 9 Insertion Local Search

1: INPUT: A linear order $a = \langle a_1, a_2, \ldots, a_n \rangle$ and values y_{ij} for all ordered pairs $(i, j) \in A^2$.
2: Set $i := 1$.
3: **while** $i < n + 1$ **do**
4: **if** $\max_{j=1,\ldots,n+1} v(a, i, j) > v(a)$ **then**
5: Set $a := \langle \ldots, a_{j-1}, a_i, a_j, \ldots \rangle$.
6: Set $i := 1$.
7: **else**
8: Set $i := i + 1$.
9: **end if**
10: **end while**
11: OUTPUT: A linear order $a = \langle a_1, a_2, \ldots, a_n \rangle$

In this local search heuristic, the neighbourhood of an order is defined as all orders that can be constructed from the current order by moving a single alternative to a different position. For a given alternative, all possible moves are evaluated. If the best possible move for this alternative improves the objective value, this move is made and the ordering updated. The algorithm terminates if there are no more improvements possible through moving a single alternative.

In Algorithm 10, we show how we combine the algorithms described so far. We denote the best ordering found so far by $\bar{a}$.

This implementation combines the constructive best insertion and local search insertion algorithms to quickly find linear orders which are good solutions to the pricing problem. Since the outcome of the constructive heuristic depends strongly on the order in which alternatives are added to the linear order, we use a multi-start procedure. For each alternative $i \in A$, we run the algorithm once, inserting i first. From these multiple

Algorithm 10 Pricing Problem

1: INPUT: A set of alternatives A and values y_{ij} for all ordered pairs $(i, j) \in A^2$.
2: **for** $i \in A$ **do**
3: Run Algorithm 8, starting with the insertion of alternative i.
4: Run local search Algorithm 9 with starting solution a.
5: If $v(a) > v(\bar{a})$, set $\bar{a} := a$.
6: **end for**
7: If $v(\bar{a}) \leq 0$, run CPLEX for an optimal solution to the pricing problem.
8: OUTPUT: $\bar{a}$

runs, we save the best solution to the pricing problem, and if this solution can be used to add a variable to the restricted master problem, we do so. If the objective value found through the heuristics is lower than or equal to 0, we have not found any variables to add to the master problem. However, this does not mean no such variables exist. We therefore use an exact solver as a back-up, which either finds a new variable, if one exists, or provides us with proof that such a variable does not exist. In this way, we are still guaranteed a correct test of the mixture model.

To further speed up the column generation algorithm, we look for multiple solutions to our pricing problem. By adding additional variables in a single iteration, we hope to get larger improvements in our master problem. In our implementation, we keep using the multi-start best insertion heuristics, saving every solution that provides an improving variable (and is different from already saved solutions). We denote the multiple saved solutions by $a^1, a^2, \ldots$.

Adjusting pricing solutions

As a final addition to the column generation algorithm in our implementation, we propose a way to adjust a solution to the pricing problem slightly, so that they match the data as closely as possible. The intuition of the

Algorithm 11 Adding multiple columns

1: INPUT: A set of alternatives A and and values y_{ij} for all ordered pairs $(i, j) \in A^2$.
2: **for all** $i \in A$ **do**
3: Run Algorithm 8, starting with the insertion of alternative i.
4: Run local search Algorithm 9 with starting solution a.
5: If $v(a) > 0$ and $a \neq a^k$, for all saved a^k, save a as a^{k+1}.
6: **end for**
7: If there does not exist a^k, with $v(a^k) \leq 0$, run CPLEX for an optimal solution to the pricing problem, else skip this step.
8: OUTPUT: All orders a^k.

adjustment we propose is as follows. Suppose a linear order with $i \succ j$ is found as a solution to the pricing program, which can be added to the master problem as an improving column. However, if $y_{ij} = y_{ji} = 0$, then a linear order with $j \succ i$ could have the same objective value in the pricing problem. Now suppose $p_{ji} = 1$. If this is the case, it is clear that an order with $i \succ j$ will never be used in the eventual solution to our master problem, and we should add an order with $j \succ i$ instead. A similar reasoning can be used if $p_{ji} > 0.5$. It may not be the case that a variable with $i \succ j$ will never be used, but is likely that the eventual solution will use more variables corresponding to orders with $j \succ i$. We therefore add the steps outlined in Algorithm 12 at the end of our heuristic pricing algorithms. The y_{ij} values of the pricing problem are changed, and another local search algorithm is run, with the goal of placing alternatives $i \in A$ with large p_{ij} values higher within the linear ordering. In particular, for every pair of alternatives $(i, j) \in A^2$ for which the ranking within the linear order has no effect on the value of the pricing solution ($y_{ij} = y_{ji} = 0$), we set $y_{ij} := p_{ij}$ and $y_{ji} := p_{ji}$. To ensure the value of the solution to the pricing solution is not lowered, every relation in the linear ordering solution which adds any value to the objective function is assigned an arbitrarily high y_{ij} value.

Algorithm 12 Adjusting the pricing problem solution

1: INPUT: A set of alternatives A, an order a and a number y_{ij}.
2: **for all** $(i,j) \in A^2, i \neq j$ **do**
3: If $i \succ j$ in a and $y_{ij} > 0$, set $y_{ij} := 1000$
4: If $y_{ij} = y_{ji} = 0$, set $y_{ij} := p_{ij}$ and $y_{ji} := p_{ji}$
5: **end for**
6: Run local search Algorithm 9, with starting solution a.
7: OUTPUT: An order a.

7.4.2 Datasets

We generated four distinct classes of datasets, which we will describe in this section.

The first two of these classes (*Inside Easy* (IE) and *Inside Hard* (IH)) were generated so as to satisfy the mixture model in the following way. First, for a given number n of alternatives, t linear orders over these alternatives were randomly generated. For the easy datasets, $t = 20$, while $t = 5$ for the hard datasets. Next, $t - 1$ random numbers (q_m) between zero and one were drawn and ranked from small to large. There are then t intervals $[0, q_1], [q_1, q_2], \ldots, [q_{t-1}, 1]$, which define t numbers $q_i - q_{i-1}$ (with $q_0 = 0$ and $q_t = 1$). Each of these t number corresponds to a different generated linear order, i.e., we set $x_m = q_m - q_{m-1}$ for $m = 1, \ldots, t$. The p_{ij} values are then set as follows, $p_{ij} = \sum_{m \in O_{ij}}^{t} x_m$.

The two other classes were generated so that they are unlikely to satisfy the mixture model, these are called *Outside Easy* (OE) and *Outside Hard* (OH). Outside Easy is generated in a simple fashion. For every pair of alternatives $(i,j) \in A^2$, with $i > j$, a number p_{ij} is drawn from a uniform distribution between zero and one and $p_{ji} = 1 - p_{ij}$. While the generation process does not guarantee non-rationalizability, it turned out that all of these datasets do violate the mixture model. This is due to the restrictive nature of these models. Monte Carlo simulation shows only about 5% of datasets containing 5 alternatives generated in this manner satisfy the mix-

ture model, and that this percentage decreases with an increasing number of alternatives (Regenwetter et al., 2010). For Outside Hard, initial p_{ij} values are drawn using the same procedure. These numbers are the input to an optimization problem, which minimizes the changes in the p_{ij} values, under the constraint that for every triple of alternatives, $i, j, k \in A$, the inequality $p_{ij} + p_{jk} + p_{ki} \leq 2.01$ ($p_{ij} + p_{jk} + p_{ki} \leq 2$ is a necessary condition for rationalizability). In this way, the resulting dataset is generally much closer to satisfying the mixture model than the p_{ij} values drawn from uniform distributions. Notice however, that a dataset which violates the conditions for rationalizability, will still violate these conditions after the procedure.

7.4.3 Results

In this section, we will discuss some results on computation times for mixture model tests. We will compare the different algorithms as described in Section 7.4.1, tested on the datasets described in 7.4.2. All datasets were generated for $n = 20$, and averages are reported over 5 datasets per generation method. All computation times reported in this chapter were obtained on a dual core 2.5 GHz computer with 4 GB RAM. The master problem is solved using CPLEX 12.4, as is the pricing problem for exact solutions.

Let us first look at a comparison of computation times using an exact integer programming solver for the pricing problem, and implementations using the best insertion heuristics. For this case, we include both the algorithm for adding the best heuristic solution found (BI), and the algorithm where all improving columns were added (BIM).

We note that the computation times decrease substantially by using heuristic methods for the pricing problem. Generally, these methods are able to find improving columns in most iterations of the pricing problem, allowing the computationally expensive exact pricing problems to be skipped.

[1] Computation time exceeded 2 hours (7200s) for 4 out of 5 datasets

Datasets	Exact	BI	BIM
Inside Easy	47.32s	10.21s	5.41s
Inside Hard	2598.48s	175.94s	182.33s
Outside Easy	20.46s	1.39s	1.15s
Outside Hard	No Results[1]	3483.82s	3287.98s

Table 7.1: *Computational results for exact pricing problem and simple heuristics.*

For many satisfying datasets, the mixture model test can be done without having to run any exact tests. For the datasets with violations, at least one exact test is necessary for a guarantee that no improving columns exist. For the outside easy datasets, no extra exact pricing problems are necessary. For the outside hard datasets however, most of the runtime is linked to exact pricing problems. Furthermore, we note the similar results for the heuristic algorithm with one and multiple added columns. An interesting observation is that while overall computation times are similar, the division over different parts of the algorithm is different. When adding multiple columns in each pricing iteration, the number of pricing iterations that are necessary decreases significantly (depending on the class of dataset, between 70% and 90%). However, the total number of added columns is much larger, which increases the time needed in updating and running the master problem.

Next we look at the impact of the adjustment of pricing solutions, given in Table 7.2. BIA represents the BI algorithm with adjustments, while BIMA represents BIM with adjustments. Adjusting the pricing solutions has a positive impact on computation times for all datasets. This reduction in computation time is, depending on the basic algorithm, due to two effects. In the simple best insertion algorithms, the adjustment of pricing solutions leads to a large decrease in the number of iterations the column generation algorithm requires. In the variant with multiple added variables in each iteration, the improvement is due to a smaller number of variables being added in each iteration.

Datasets	BI	BIA	BIM	BIMA
IE	10.21s	7.69s	5.41s	5.37s
IH	175.94s	87.41s	182.33s	181.84s
OE	1.39s	1.30s	1.15s	0.71s
OH	3483.82s	3028.06s	3287.98s	2873.71s

Table 7.2: *Computational results including adjustments to the pricing solutions.*

Finally, we look at including the stronger stopping condition (BIA SC and BIMA SC) discussed in Section 7.3.4 and report results for the hard non-consistent datasets in Table 7.3. Clearly, this more stringent stopping condition quickly identifies datasets which can not be consistent with the mixture model, cutting computation times from almost one hour to less than one minute on average. Due to the stronger stopping condition for pricing problem solutions, the proof that the mixture model cannot be satisfied is usually obtained in the first iteration that uses exact methods.

Datasets	BIA	BIA SC	BIMA	BIMA SC
OH	3028.06s	52.63s	2873.71s	33.56s

Table 7.3: *Computational results including adjustments to the pricing solutions.*

7.5 Bayes Factor Calculation

As we noted in several previous chapters, tests of models of behaviour are generally sharp. Either the data satisfy the model or they don't. The same is true for the model we test in the current chapter. However, given that we assume stochastic decision making, such sharp tests create additional difficulties. Indeed, suppose we observe that an alternative i is chosen over another alternative j only 1 out of 10 times. If we test the mixture model on the data, we can use $p_{ij} = 0.1$ and we may find the model is

rejected. However, suppose that for $p_{ij} \geq 0.15$ the model is satisfied. If the decision maker prefers i over j 15% of the time, then it is still likely that he only picks i once over 10 trials and we "observe" a violation. Observed behaviour can thus fail to satisfy Condition (7.1) even though the decision maker satisfies the model.

In this section, we will look at the *Bayes Factor* as a way for accounting for this problem (Klugkist and Hoijtink, 2007). Informally speaking, given observed behaviour, a *posterior distribution* can be calculated, which represents how likely specific choice probabilities are to generate the observed data. From this posterior distribution, we draw samples and test whether these satisfy the mixture model. The percentage of such sampled datasets which satisfy the mixture model (out of the total number of tested samples), provides an approximation of the posterior probability that the decision maker satisfies the mixture model, given the observed choices. To be able to compare models, this posterior probability can be compared against the percentage of samples from a prior distribution that are consistent.

In line with Cavagnaro and Davis-Stober (2014), we take the prior distribution of a p_{ij} value to be a uniform distribution between 0 and 1. The percentage of samples consistent with the mixture model is then equal to the volume of the linear ordering polytope compared to the unit hypercube. We are mainly interested in computing the posterior probability. The posterior distributions of the p_{ij} values is given by Beta distributions. In particular, let q_{ij} be the number of times i is observed chosen in a choice between i and j. Then, for every pair of alternatives $(i, j) \in A^2$, the distribution of p_{ij} is given by Beta $= (q_{ij} + 1, q_{ji} + 1)$. Given a sampled p_{ij} value for every pair, we have a synthetic dataset for which the mixture model can be tested.

To estimate the posterior probability as closely as possible, large numbers of these synthetic datasets must be tested for consistency with the mixture model. However, since all datasets are sampled from the same distribution,

these datasets are usually quite similar. In this section, we will look at some ways the column generation procedure described in this chapter can exploit these similarities to quickly test many datasets. First, we will look at how re-using the columns generated in one test as a starting set for new datasets speeds up these tests in subsection 7.5.1. Next, in subsection 7.5.2 we show that the objective function of the final pricing problem of a rejected dataset provides inequalities, which are necessary conditions for rationalizability by the mixture model. These inequalities can be used to quickly test whether datasets violate the mixture model.

7.5.1 Starting Sets

As all generated datasets are fairly similar, it is likely that the optimal solutions to the linear programs (7.2-7.5) for these datasets use many of the same variables. By testing the points sequentially and using the variables generated while testing previous datasets as a starting set for new samples, we attempt to minimize the number of pricing iterations needed. Table 7.4 demonstrates that this is indeed works. For a given dataset of observed choices with 10 alternatives, we calculated the posterior Beta distributions. From samples from these distributions, we created 10000 simulated datasets. Of these datasets, approximately 10% satisfy the mixture model. The table clearly shows that when testing these datasets, tests of the first datasets require a large number of variables to be generated. More than 20% of the total number of variables is generated in the first 1% of the tests. The number of variables generated then quickly tails off, as the starting sets for later tests are generally sufficient to prove (non-)consistency with the mixture model.

Sampled Dataset	1	2	10	100	1000	2500	5000	10000
Total # Variables	66	113	172	430	986	1275	1587	1892

Table 7.4: *Cumulative number of variables.*

The re-use of variables also has a large effect on the necessary computation times. For this test, we used only exact algorithms for the pricing problem. Using these exact algorithms, testing a single point from a minimal starting set generally takes around 10 seconds. Most of this computation time is due to the pricing problems. Given a starting set with all variables necessary for an optimal solution to the master problem, this time drops to either about 0.05 seconds (in the case of satisfying datasets) or 0.15 (in the case of non-satisfying datasets). The longer time needed for non-satisfying datasets is due to the fact that these datasets require at least one pricing problem to be solved to prove that an optimal solution to the master problem has been found. All 10000 generated datasets could be tested in 2333.61 seconds, or about 0.25 seconds on average.

7.5.2 Valid Inequality Pool

Using the starting sets as described in the previous subsection offers significant speed-ups for testing many similar datasets, but one problem that remains is that for each non-satisfying dataset, a pricing problem must be run to prove that the optimal solution to the master problem has been reached. Since pricing problems are a significant factor in overall computation times, we wish to avoid this if possible. One way to do so is as follows. In case the pricing problem fails to find a new variable for the master problem, we have identified an inequality that must be satisfied by any dataset consistent with the mixture model. Thus, from every non-rationalizable dataset we obtain an inequality. Before using the column generation algorithm on further datasets, we can first test whether these datasets violate any of the known inequalities. If violations are found, we can conclude the dataset does not satisfy the mixture model and no test using column generation is necessary.

Theorem 7.5.1. *Suppose there exist numbers y_{ij} for all $(i,j) \in A^2, i \neq j$, and a number c, such that there does not exist a linear order $m \in O$ for which $\sum_{ij:i\succ_m j} y_{ij} > c$. Then there exists no dataset with numbers p_{ij}*

for all $(i,j) \in A^2, i \neq j$*, that satisfies the mixture model and for which* $\sum_{i,j \in A} y_{ij} p_{ij} > c$*.*

Proof. This proof is analogous to the proof for Theorem 7.3.1. □

Using these valid inequalities, the total computation time of the 10000 generated datasets drops from 2333.6 seconds to 408.4 seconds. This decrease in computation time is mainly due to the number of pricing problems that we avoid. In total, 9030 datasets are non-rationalizable, but only 119 of these datasets required solving pricing problems to prove this. The other 8911 datasets violated one (or more) of the valid inequalities identified solving earlier datasets. Since checking the list of valid inequalities usually takes less than 0.001 second, this is a large improvement compared to the 0.15 seconds for solving an average pricing problem.

7.6 Conclusion

In this chapter, we have presented an algorithm for testing models of stochastic preferences (mixture models), based on column generation. This algorithm is capable of handling datasets of such size that the number of linear orders over all alternatives, and thus the number of variables in (7.1) would make the system of equalities prohibitive to solve. We propose several ways of speeding up the algorithm, by making use of heuristics, particular choices about the variables to be added if there are multiple candidates and by proving the pricing solutions must reach certain thresholds for the system of equalities (7.1) to be satisfied. Furthermore, we show that the column generation algorithm is well-suited for testing large numbers of similar datasets, as variables can be re-used and the pricing objective function provides valid inequalities.

Chapter 8

Recognizing Single-Peaked Preferences on Aggregated Choice Data

8.1 Introduction

In the previous chapter, we studied a stochastic model of general preferences, i.e., for a set of alternatives, a decision maker can hold any preference ordering. In this chapter, we will study a special case of this model, where preferences are restricted in some sense. An important such restriction is given by *single-peakedness*, introduced by Black (1948). Suppose a linear ordering exists, that ranks all alternatives along a line. An agent's preferences are then single-peaked if she has a most preferred alternative, the *peak*, and when comparing two alternatives that are both on the same side of the peak, the alternative closest to the peak is preferred. This restriction is very natural when considering a situation where a single attribute of the alternatives drives the choice, for example, an election where candidates range from left wing to right wing, or choices over budgets of various sizes. Given these examples, it is no wonder that this restriction has gained central importance in the areas of political science and social choice. Apart

from being an appealing model in these areas, the assumption of single-peaked preferences has led to interesting theoretical results. For example, aggregation of single-peaked preferences avoids the Condorcet-paradox.

Within a social choice context, single-peaked preferences (and closely related preference restrictions), have been studied extensively. In contrast to our setting however, this literature does not use aggregated preference data, instead assuming the full preference profiles of decision makers are known. We note work by Bartholdi III and Trick (1986), who provide a polynomial time algorithm to test whether the observed preference profiles are single-peaked with respect to some ordering of the alternatives and to identify this ordering. Escoffier et al. (2008) provide a different algorithm for the same problem with a better worst-case bound. Ballester and Haeringer (2011) give two forbidden substructures, whose absence is a necessary and sufficient condition for the given preference profile to be consistent with single-peakedness. Furthermore, Trick (1989) provides an algorithm for recognizing single-peakedness on trees, which again runs in polynomial time. Finally, Doignon and Falmagne (1994), Knoblauch (2010) and Elkind and Faliszewski (2014) investigate a closely related preference restriction, one-dimensional Euclidean preference profiles, and all provide polynomial time algorithms.

The main contributions of this chapter are as follows.

- Given an ordering of the alternatives, we provide necessary and sufficient conditions for testing whether aggregated preferences are consistent with a mixture model of single-peaked preferences. These conditions can be tested in polynomial time.
- We provide a polynomial time algorithm which given the aggregated preferences, provides an ordering of the alternatives for which a mixture model of single-peaked preferences is satisfied (if such an ordering exists).

The rest of this chapter is organized as follows. In section 8.2, we further define single-peaked preferences and the mixture model. Section 8.3 contains our main result, necessary and sufficient conditions for a mixture model of single-peaked preferences to hold. Next, section 8.4 provides further results, specifically two algorithms to identify the underlying ordering of the alternatives. Finally, section 8.5 concludes.

8.2 Notation and Definitions

Consider a set A, consisting of n alternatives, and a dataset $P = \{p_{ij} \geq 0, \forall (i,j) \in A^2, i \neq j\}$. The values p_{ij} represent the probability that i is chosen over j. As we assume strict preferences, $p_{ij} + p_{ji} = 1$. We also consider a (given) ordering of alternatives in A. This ordering is complete, asymmetric and transitive and is denoted by $\gg$. Furthermore, we consider preference orderings over all alternatives. We will use the index m to denote a particular preference ordering. If for a given preference ordering m, an alternative i is preferred over another alternative j, we denote this by $i \succ_m j$.

Definition 8.2.1. *A preference ordering m is single-peaked with respect to a given ordering of the alternatives $\gg$ if and only if for every triple $(i,j,k) \in A^3$ we have:*

$$\text{if } (i \gg j \gg k \text{ and } i \succ_m j) \text{ then } i \succ_m k. \tag{8.1}$$

$$\text{if } (i \gg j \gg k \text{ and } k \succ_m j) \text{ then } k \succ_m i. \tag{8.2}$$

The set of all preference orderings that are single-peaked with respect to an ordering $\gg$ is denoted by $O^{\gg}$. We further consider the subsets $O_{ij}^{\gg}$, defined as follows: $m \in O_{ij}^{\gg}$ if both $m \in O^{\gg}$ and $i \succ_m j$. A mixture model of preference assumes that, a decision maker has a number of different preference orderings, each with an associated probability. When faced with a choice between alternatives i and j, the probability that the decision maker chooses i is equal to the sum of the probabilities of all preference orderings in which i is preferred over j.

Definition 8.2.2. *A dataset P can be rationalized by a mixture model of single-peaked linear ordering preferences with respect to a given ordering of alternatives $\gg$ if and only if there exist numbers $x_m \geq 0, \forall m \in O^{\gg}$ for which:*

$$\sum_{m \in O^{\gg}_{ij}} x_m = p_{ij}, \qquad \forall (i,j) \in A^2, i \neq j. \tag{8.3}$$

8.3 Consistency conditions

We claim that the existence of a solution to the system of equalities (8.3) can be checked easily by verifying a condition on the p_{ij} values. We will prove both the sufficiency and necessity of this condition, and then finish this section by showing that the condition may be tested in polynomial time.

Theorem 8.3.1. *A dataset P can be rationalized by a mixture model of single-peaked preferences with respect to a given ordering $\gg$ if and only if for every triple $(i,j,k) \in A^3$ we have:*

$$\text{if } i \gg j \gg k \text{ then } p_{ij} \leq p_{ik} \text{ and } p_{kj} \leq p_{ki}. \tag{8.4}$$

Before embarking on a proof of this theorem, let us first make a couple of observations. First, we note that the condition (8.4) is similar, but subtly different from the conditions for Robinsonian dissimilarities (Robinson, 1951). The main differences are as follows, for dissimilarities the values p_{ij} are symmetric, i.e, $p_{ij} = p_{ji}$ (and there is no constraint on $p_{ij} + p_{ji}$). Furthermore, a dissimilarity is Robinsonian with respect to an order if and only if for every triple $(i,j,k) \in A^3$, with $i \gg j \gg k$, it must be the case that $p_{ij} \leq p_{ik}$ and $p_{jk} \leq p_{ik}$. We also note that condition (8.4) is a reformulation of conditions (8.1)-(8.2) for the setting with aggregated preferences. While we will formally argue the necessity later on, it is clear that if condition (8.4) is violated, at least part of the population has to hold preferences that violate either condition (8.1) or (8.2). However, the

sufficiency of this condition is not so straightforward. Indeed, if we look at mixture models with general preferences, we find that the list of necessary and sufficient conditions is exponential in the number of alternatives[1]. This is the case, even though general preference orderings are constrained only by transitivity, which can also be defined by a condition over all triples. To prove the sufficiency of condition (8.4) we will proceed as follows. First, we will describe an algorithm whose goal it is to find single-peaked preference orders $m \in O^{\gg}$ and associated values x_m satisfying (8.3). We will show this algorithm is able to do so if the dataset P satisfies condition (8.4).

The complete pseudo-code is given in Algorithm 13, here we will give a short overview. The main idea is that if we know the aggregated preferences of (a part of) a population, we wish to identify a single-peaked preference order held by a part of that population, explaining a portion of the observed preferences. This usually leaves some part of the data unexplained, captured in the algorithm by the variables $\tilde{p}_{ij}$. For this unexplained data, another single-peaked preference order is then found, and so on. These orders are constructed by iteratively adding alternatives, starting from the most preferred to the least preferred. To do this, we first identify the set of alternatives which can be added to the order (I). We define the set M as the set of alternatives which have already been added to the order, obviously if $i \in M$, then $i \notin I$, as alternatives can only be in the order once. Furthermore, if there exists an alternative $i \notin M$, for which there is some other alternative $j \notin M$ such that $\tilde{p}_{ij} = 0$, this alternative i can not be added to the order before j is, as no member of the population can hold such a preference ordering, thus $i \notin I$. From this set I, we then choose the alternative which is first in the underlying ordering $\gg$ and add

[1] Suck shows that testing the mixture model for strict linear preferences is equivalent to testing membership of the linear ordering polytope (Suck, 1992). A full facet description of this polytope thus gives necessary and sufficient conditions for the mixture model of strict linear preferences. However, since separation over this polytope is NP-Hard, the facet description is exponential in the number of alternatives. Full descriptions are only known for small number of alternatives (Fiorini, 2006; Martí and Reinelt, 2011)

it to the ordering $\succ_m$. These steps are repeated until all alternatives are ranked.

The process of constructing single-peaked linear orders is the key part of the algorithm and is found in the loop (4-10). For this loop we will prove four properties, which all depend on the condition (8.4) being satisfied for $\tilde{p}_{ij}$. First, that the loop always runs to completion, i.e. it outputs a strict linear order $\succ_m$. Second, that $\succ_m$ is single-peaked with regards to $\gg$. Third, that this $\succ_m$ can be given a weight x_m, and for all $(i,j) \in A^2$ for which $i \succ_m j$, we have $x_m \leq \tilde{p}_{ij}$ and that there exist some $i,j \in A$ for which $i \succ_m j$ and $x_m = \tilde{p}_{ij}$. Finally, at the end of each loop, the values of $\tilde{p}_{ij}$ satisfy condition (8.4). Given these four properties, we will be able to prove that the algorithm provides single-peaked linear orders $m \in O^{\gg}$ and values x_m that satisfy (8.3).

Algorithm 13 Finding Single-Peaked Preferences

1: INPUT: p_{ij} for all $(i,j) \in A^2$ and $\gg$.
2: Set $\tilde{p}_{ij} := p_{ij}$ for all distinct $i,j \in A$, $m := 1$ and create $\succ_m := \varnothing$, $M := \varnothing$ and $I := \varnothing$.
3: **while** $\tilde{p}_{ij} + \tilde{p}_{ji} > 0$ for all distinct $i,j \in A$ **do**
4: **for** $|M| < |A|$ **do**
5: Set $I := \{i \in A \backslash M : \tilde{p}_{ij} \neq 0, \forall j \in A \backslash M, j \neq i\}$
6: If $I = \varnothing$, STOP.
7: Set $i^* := i$ with $i \in I$ for which $\forall j \in I, j \neq i : i \gg j$.
8: $\forall j \in M$, set $j \succ_m i^*$
9: Set $M := M \cup \{i^*\}$
10: **end for**
11: Set $x_m := \min_{i,j \in A : i \succ_m j} \tilde{p}_{ij}$.
12: Set $\tilde{p}_{ij} := \tilde{p}_{ij} - x_m, \forall i,j \in A$ for which $i \succ_m j$.
13: Set $m := m + 1$
14: Set $M := \varnothing$
15: **end while**
16: OUTPUT: For all $i \in \{1, \ldots, m\}$ a value x_i and order $\succ_i$.

Claim 8.3.1. *If the values $\tilde{p}_{ij}$ meet condition (8.4), the loop (4-10) will return a linear order.*

Proof. If for some M there does not exist an $i \in A\backslash M$ such that $\tilde{p}_{ij} > 0, \forall j \in A\backslash M$, the algorithm will halt in line 6 without constructing an order. We argue by contradiction : suppose this is the case and the condition (8.4) is satisfied. Now consider $i \in A\backslash M$ with $i \gg j$ for all $j \in A\backslash M$. There is some j for which $i \gg j$ and $\tilde{p}_{ij} = 0$. Now let i' be the immediate neighbour of i, i.e., there is no $k \in A\backslash M$ such that $i \gg k \gg i'$. Then by condition (8.4), we have $\tilde{p}_{ii'} = 0$. As $\tilde{p}_{ii'} + \tilde{p}_{i'i} > 0$ and thus, $\tilde{p}_{i'i} > 0$, (8.4) further implies $\tilde{p}_{i'l} > 0, \forall l \in A\backslash M$ for some $l \gg i'$. Furthermore, for i', there also exists some $j \in A\backslash M$ for which $\tilde{p}_{i'j} = 0$, this j must have $i' \gg j$. By the same argument as for i, we can see that $\tilde{p}_{i'i''} = 0$ and so on until we reach the alternative n, for which $j \gg n, \forall j \in A\backslash M$. This alternative n has $p_{nj} > 0, \forall j \in A\backslash M$, a contradiction. Given that condition 8.4 holds, there must exist an alternative which can be added to M in each step of the for loop, and the algorithm finds a strict linear order.

□

Claim 8.3.2. *If the values $\tilde{p}_{ij}$ meet condition (8.4), the linear order returned by the loop (4-10) is single-peaked with respect to $\gg$.*

Proof. First, we note that the set I has the following property. For each pair of alternatives $i, j \in I$, there does not exist a $k \notin I$ and $k \in A\backslash M$, for which $i \gg k \gg j$. This can be argued by contradiction. Suppose such a k exists, then there also exists an alternative $l \in A\backslash M$ for which $\tilde{p}_{kl} = 0$. Without loss of generality we assume $l \gg k$. By conditions (8.4), $\tilde{p}_{lj} \geq \tilde{p}_{lk}$ and thus also $\tilde{p}_{jl} = 0$ in which case $j \notin I$.

Furthermore, consider an alternative j. In a given iteration of the loop, we have $j \in A\backslash M$, $j \notin I$, and $j \gg i$ for all $i \in I$. Only if there does not exist an alternative $j' \in A\backslash M$, $j' \notin I$, and $j \gg j' \gg i$, is $j \in I$ possible in the next iteration. Again we argue by contradiction, if $j \notin I$ in one iteration and $j \in I$ in the next, an alternative $i \in I$ with $\tilde{p}_{ji} = 0$ was added to M.

If j' exists, condition (8.4) implies $0 = \tilde{p}_{ji} > \tilde{p}_{jj'}$ and j can not be added to I. The same argument applies for $i \gg j$

From these two observations the claim easily follows. Suppose $i \in A$ is the first alternative set for order m. For every pair of alternatives $j, k \in A$, for which $j \gg k \gg i$, k is added to the order m before j, as the iteration in which $j \in I$ must be after the iteration in which $k \in I$ and k will be added to the order immediately when $k \in I$. For every pair of alternatives $h, l \in A$, for which $i \gg h \gg l$, h is added to the order before l as $l \in I$ implies $h \in I$ and if both $l, h \in I$, l can not be chosen as $h \gg l$ by construction in line 7 □

Claim 8.3.3. *In line 11, x_m is set such that $x_m \leq \tilde{p}_{ij}$ for all $(i,j) \in A^2$ for which $m \in O_{ij}^{\gg}$ and there exists some $i, j \in A$ for which $m \in O_{ij}^{\gg}$, such that $x_m = \tilde{p}_{ij}$ and $x_m > 0$.*

Proof. This is true by construction, an alternative i is only added to M if $\forall j \in A \backslash M, \tilde{p}_{ij} > 0$. As $i \succ_m j$ is only the case if j was added to M after i, then all $\tilde{p}_{ij}$ over which the minimization are done are strictly positive. By nature of the minimization, there is also at least one $\tilde{p}_{ij}$ to which x_m is equal and x_m is no larger than any of the $\tilde{p}_{ij}$. □

Claim 8.3.4. *If condition (8.4) is satisfied at the beginning of the loop (3-15), the $\tilde{p}_{ij}$ values will satisfy condition (8.4) at the end of the loop.*

Proof. In this proof, we will denote the value $\tilde{p}_{ij} + \tilde{p}_{ji}$ by y. Throughout, we will assume that condition (8.4) is satisfied in line 3. First, let us consider the situation $i \gg j \gg k$, as (8.4) holds, $\tilde{p}_{ij} \leq \tilde{p}_{ik}$. Only if an order m exists such that $j \succ_m i \succ_m k$ is found, will $\tilde{p}_{ik}$, but not $\tilde{p}_{ij}$, decrease in line 11. If both $i, j \in I$, i will be added to m first due to line 7. Thus, $j \succ_m i$ implies that there exists $l \in A$, such that $\tilde{p}_{il} = 0$ and $\tilde{p}_{jl} > 0$. We will consider three distinct situations. First, $l \gg i$, then $j \gg l$ and finally $i \gg l \gg j$. Let us consider $l \gg i$. As $\tilde{p}_{il} = 0$, $\tilde{p}_{li} = y$, which implies $\tilde{p}_{lj} = y$ and $\tilde{p}_{jl} = 0$ as $l \gg i \gg j$ demands $\tilde{p}_{li} \leq \tilde{p}_{lj}$. Therefore, l would prevent

both i and j from being added to I and m. As soon as l was added to M, both $i \in I$ and $j \in I$ are possible, and again i would be added to m before j. $l \gg i$ thus can not lead to $j \succ_m i$. In the case of $j \gg l$, it is clear that because $\tilde{p}_{il} = 0$, we must also have $\tilde{p}_{ij} = 0$, therefore $\tilde{p}_{ij} \leq \tilde{p}_{ik}$ can not be violated. Finally, if $i \gg l \gg j$, we must have $j \succ_m l$, if this were not the case i could be added to m after l but before j. By the earlier arguments in this paragraph $j \succ_m l$ while $l \gg j$ is only possible if there is some other alternative $l' \in A$, with $l \gg l'$ and $\tilde{p}_{ll'} = 0$. $j \gg l'$ gives $\tilde{p}_{lj} = 0$ and therefore $\tilde{p}_{ij} = 0$. If on the other hand $l \gg l' \gg j$, we can repeat the same argument until we find some l'' with $i \gg l'' \gg j$ and $\tilde{p}_{jl''} = y$, implying $\tilde{p}_{ji} = y$ and $\tilde{p}_{ij} = 0$. In conclusion, if $i \gg j \gg k$, we can only have $j \succ_m i \succ_m k$ if $\tilde{p}_{ij} = 0$. If this is the case, then $\tilde{p}_{ij} \leq \tilde{p}_{ik}$ is satisfied, as $\tilde{p}_{ik} \geq 0$.

The second situation is $k \gg j \gg i$, in which case we also have $\tilde{p}_{ij} \leq \tilde{p}_{ik}$. Here, only an order with $j \succ_m i \succ_m k$ can lead to the condition being violated in line 15. In the previous paragraph, we established that if $a \gg b$ and the algorithm places $b \succ_m a$, we have $\tilde{p}_{ab} = 0$. Here, $k \gg i$ and $i \succ_m k$, so $\tilde{p}_{ki} = 0$. As $\tilde{p}_{ki} = 0$, it must be the case that $\tilde{p}_{ik} = y$ and thus $\tilde{p}_{ij} \leq y = \tilde{p}_{ik} = y$.

□

We are now in a position to prove Theorem 8.3.1.

Proof. First, we prove sufficiency of the condition. We have shown, by combining claims 8.3.1 and 8.3.2, that given a set of values $\tilde{p}_{ij}$ which satisfy condition (8.4), we can find a strict single-peaked linear order. By claim 8.3.3 we have also seen that we can attach a weight to this order which is non-negative. Even stronger, we have shown that this weight is equal or less than the value $\tilde{p}_{ij}$ for some pair $(i, j) \in A^2$, for which $x_m \in O^{\gg}_{ij}$. As the final step of the loop will decrease these $\tilde{p}_{ij}$ values, at least one of these values is set to zero in each run. After at most $O(n^2)$ iterations of the loop, each value $\tilde{p}_{ij}$ will then be zero. It can be easily

checked that at this point, the values x_m form a solution to (8.3). As this proof requires the loop to be run multiple times, and the loop requires condition (8.4) to hold, claim 8.3.4 is crucial, as it shows that if the input of the loop satisfies the condition, the output will as well.

Next, we turn to the necessity of condition (8.4). This can easily be verified by a three alternative example. Suppose $(i, j, k) \in A^3$, with $i \gg j \gg k$. By definition of single-peaked linear orders, each order for which $i \succ j$ also has $i \succ k$. This means $O^{\gg}_{ij} \subset O^{\gg}_{ik}$ and $\sum_{m \in O^{\gg}_{ij}} x_m \leq \sum_{m \in O^{\gg}_{ik}} x_m$. A solution to (8.3) requires $p_{ij} = \sum_{m \in O^{\gg}_{ij}} x_m$ and $p_{ik} = \sum_{m \in O^{\gg}_{ik}} x_m$, and this proves $p_{ij} \leq p_{ik}$. The same argument can be used for $p_{kj} \leq p_{ki}$. This shows necessity of the condition. □

Theorem 8.3.2. *For a given dataset P and ordering $\gg$, Condition (8.4) can be checked in time $O(n^2)$.*

Proof. It can be easily seen that this condition may be checked in polynomial time. As written, two inequalities must be checked for each triplet of alternatives, giving an obvious $O(n^3)$ time test. This can be improved upon by noting that when using a matrix of p_{ij} values, with rows and columns ranked according to the ordering $\gg$, values above the diagonal must be non-decreasing in the rows and the columns. Conversely, as $p_{ij} + p_{ji} = 1$, values below the diagonal are non-increasing in both rows and diagonals (In fact, the lower triangle of the matrix is double graded). As such, each p_{ij} value must be compared with only two other values, providing an $O(n^2)$ test. □

8.4 Recognizing single-peaked orderings

In the previous section, we have given necessary and sufficient conditions for the data to be consistent with a mixture model of single-peaked preferences with respect to an order $\gg$. In this section, we investigate the

case where this order is not given a priori. We show that, if there do exist orderings $\gg$ for which the dataset P satisfies the mixture model, we can identify these. We note that there also exist algorithms for recognizing Robinsonian dissimilarities, which is a problem similar to the one we handle in this section. Préa and Fortin (2014) describe an algorithm to recognize Robinsonian dissimilarities in $O(n^2)$ time.

We define the set of orders L_P, with $\gg \in L_P$ if and only if P satisfies condition (8.4) with respect to $\gg$. In this section, we will prove that we can identify L_P. Note that if an order $\gg \in L_P$, the reverse order $\ll \in L_P$. This can be easily checked because the condition (8.4) depends on the relative ordering of alternatives, but not its orientation. $i \gg j \gg k$ and $i \ll j \ll k$ lead to the same constraints on P. Initially, we make the assumption that there is no subset $A' \subset A$ with $|A'| > 1$, for which the following holds: $\forall i, j \in A', k \in A \backslash A', p_{ik} = p_{jk}$. In words, this means that there is no subset of A with two or more items for which all items seem identical when compared to items outside of this subset. We will call these *Nearly-Identical Alternative Subsets* (NIA Subsets) and show how to recognize and handle such subsets in subsection 8.4.1.

We start with the special case without any NIA subsets and proceed as follows. First, we derive a number of necessary conditions for all $\gg \in L_P$. If satisfied, the second and third of these conditions, given in Claim 8.4.2 and 8.4.3, can be used to identify an *extreme alternative*, $\bar{a}$, which is either the first or last element of any ordering $\gg \in L_P$. This extreme alternative is then used as input for Algorithm 14 and we will show that, if $L_P \neq \varnothing$, this algorithm has an ordering $\gg \in L_P$ as output. We begin by deriving another necessary condition on the p_{ij} values, which we will use in further proofs.

Claim 8.4.1. *For any $\gg \in L_P$ and each triple of alternatives $(i, j, k) \in A^3$:*

$$\text{if } i \gg j \gg k \text{ then } p_{ij} \leq p_{jk} \tag{8.5}$$

Proof. Suppose this is not the case and $p_{jk} < p_{ij}$. Due to condition (8.4), we further have $p_{jk} < p_{ij} \leq p_{ik}$ and $p_{kj} \leq p_{ki}$. Equivalently, $1-p_{jk} \leq 1-p_{ik}$ or $p_{jk} \geq p_{ik}$, which violates condition (8.4). □

The next claim gives a further necessary condition for L_P to be non-empty. Specifically, L_P can only be non-empty if all alternatives are indistinguishable from each other (8.6), or if there exists an extreme alternative $\bar{a}$ which is a Condorcet-loser (8.7).

Claim 8.4.2. *Suppose there exists an ordering $\gg \in L_P$. Then either*

$$p_{ij} = 0.5, \forall (i,j) \in A^2, i \neq j, \tag{8.6}$$

or

$$\exists \bar{a} : \{p_{\bar{a}i} \leq 0.5, \forall i \in A \text{ and } \exists j \in A : p_{\bar{a}j} < 0.5\}. \tag{8.7}$$

Proof. First, suppose there is no extreme alternative $\bar{a}$ for which for all $i \in A$ $p_{\bar{a}i} \leq 0.5$. Without loss of generality, we say $p_{a_1 i} > 0.5$. Then, because $a_1 \gg i \gg a_n$ and condition (8.4), we have $p_{a_1 a_n} > 0.5$, thus $p_{a_n a_1} < 0.5$ and again by condition (8.4), we have for all $j \in A$, $p_{a_n j} < 0.5$. Thus, there is certainly at least one extreme alternative for which for all $i \in A$, $p_{\bar{a}i} \leq 0.5$.

Now suppose an extreme alternative $\bar{a}$ exists for which for all $i \in A$, $p_{\bar{a}i} \leq 0.5$, but there exists no extreme alternative for which there exists an $i \in A$ for which $p_{\bar{a}i} < 0.5$. Without loss of generality we assume that this is the case for a_1, thus for all $i \in A$, $p_{a_1 i} = 0.5$. This includes $p_{a_1 a_n} = 0.5$, so by condition (8.4), we also have $p_{a_n i} \leq 0.5$ for all $i \in A$. As we assume there is no extreme alternative for which there exists an $i \in A$ for which $p_{\bar{a}i} < 0.5$, we also have $p_{a_n i} = 0.5$ for all $i \in A$. Now consider $i, j \in A$, with $a_1 \gg i \gg j \gg a_n$. Then condition (8.5) and $p_{a_1 i} = 0.5$ imply $p_{ij} \geq 0.5$ and the same condition and $p_{a_n j} = 0.5$ imply $p_{ji} \geq 0.5$. As a result, for each pair $i, j \in A$, we have $p_{ij} = 0.5$. □

Condition (8.7) shows a way to identify an extreme alternative $\bar{a}$ for any ordering $\gg \in L_P$. However, it remains to be shown that no non-extreme alternatives share this characteristic, which we do in the next Claim.

Claim 8.4.3. *For any ordering $\gg \in L_P$, there is no non-extreme alternative a for which for all $i \in A$, $p_{ai} \leq 0.5$ holds and there exists $j \in A$ for which $p_{aj} < 0.5$.*

Proof. Suppose such an alternative exists. Without loss of generality, suppose a_1 is an extreme alternative satisfying the condition in Claim 8.4.2 and $a_1 \gg a \gg a_n$. Then $p_{a_1 a} = 0.5$ and $p_{a_n a} \geq 0.5$. By condition 8.5, we then have for all $i \in A$ for which $a_1 \gg a \gg i$, $p_{ai} \geq p_{a_1 a} = 0.5$ and for all $j \in A$ for which $j \gg a \gg a_n$, $p_{ai} \geq p_{a_n a} \geq 0.5$. Thus, such an alternative cannot exist. □

We are now in a position to describe our algorithm for identifying the orders $\gg \in L_P$. As an initial step, we will check the conditions described in Claim 8.4.2. If neither (8.6) nor (8.7) holds, $L_P = \varnothing$. Furthermore, it is easy to see that if condition (8.6) holds, P satisfies the mixture model with respect to any ordering $\gg$. In other words, L_P is the set of all linear orders over the alternatives. Finally, if an alternative $a \in A$ matching condition (8.7) is found, Claims 8.4.2 and 8.4.3 prove it is an extreme alternative $\bar{a}$ for any $\gg \in L_P$. With $\bar{a}$ as input, Algorithm 14 can now be used to identify a complete ordering $\gg \in L_P$, provided such an ordering exists.

The main idea of Algorithm 14 is as follows. Given an extreme alternative $\bar{a}$ and two alternatives $i, j \in A$ for which $p_{\bar{a}i} \neq p_{\bar{a}j}$, the relative ordering of i and j is determined, such that Condition (8.4) is satisfied for the triple $\bar{a}, i, j$. When this is done for every pair of alternatives, this results in a (partial) order of the alternatives. For any pair of alternatives, for which $p_{\bar{a}i} = p_{\bar{a}j}$, a third alternative $k \in A$ is sought, which in the partial order has $k \gg i, j$ or $k \ll i, j$ and for which $p_{ki} \neq p_{kj}$. In this way, the partial order is refined until a full ordering of the alternatives is found.

Algorithm 14 Ordering Algorithm

Input: Dataset P, set A, extreme alternative a_1.
Create ordering $\gg$.
For every $i \in A \backslash \{a_1\}$, set $a_1 \gg i$.
For each pair $i, j \in A$, for which $p_{a_1 j} > p_{a_1 i}$, set $i \gg j$.
Divide all $i \in A$ into sets $A_1, A_2, \ldots$, such that for all $i, j \in A_k$ neither $i \gg j$ or $i \ll j$.
repeat
    Find a set A_k with $|A_k| > 1$, for which there exist $i \in A \backslash A_k$ and $j, j' \in A_k$ such that $p_{ij} > p_{ij'}$.
    if $i \gg j$ **then**
        For each pair $j, j' \in A_k$ for which $p_{ij} > p_{ij'}$, set $j' \gg j$.
    else
        For each pair $j, j' \in A_k$ for which $p_{ij} > p_{ij'}$, set $j \gg j'$.
    end if
    if There exists a pair $j, j' \in A_k, j \gg j'$ and $i \in A \backslash A_k$, for which Condition (8.4) is violated **then**
        Stop.
    end if
    Divide all $i \in A$ into sets $A_1, A_2, \ldots$, such that for all $i, j \in A_k$ neither $i \gg j$ or $i \ll j$.
until For all $i = 1, \ldots, n$, $\left|A^i\right| = 1$.
OUTPUT: An ordering of the alternatives $\gg$.

Claim 8.4.4. *If there are no Nearly-Identical Alternative Subsets, Algorithm 14 terminates.*

Proof. First, note that if there exists a set A_k with $|A_k| > 1$, for which there exist $i \in A \backslash A_k$ and $j, j' \in A_k$ such that $p_{ij} > p_{ij'}$, a (partial) ordering is made of the alternatives in this set and A_k is thus further split up in line (16). As a result, if such a set is always found at the start of the loop (6-17), the stopping condition for this loop (line 17) is reached in a finite number of iterations. Now suppose Algorithm 14 does not terminate, then there is a set A_k with $|A_k| > 1$, and there exists no $i \in A \backslash A_k$, such that there exist $j, j' \in A_k$ for which $p_{ij} \neq p_{ij'}$. By definition, this set is a nearly-identical alternative subset. □

Claim 8.4.5. *If there are no Nearly-Identical Alternative Subsets, an ordering $\gg \in L_P$ exists and the alternative a_1 is an extreme alternative of this ordering, then Algorithm 14 terminates with output $\gg$ and $\gg \in L_P$.*

Proof. Given that an ordering $\gg \in L_P$ exists and a_1 is an extreme alternative, it must be the case that $a_1 \gg i$ for all $i \in A$. Furthermore, it can be easily checked than whenever the relative ordering of two alternatives $i, j \in A$ is fixed in relation to a third alternative $k \in A$ (say $k \gg i \gg j$), whether in line (4) or the loop (6-17), the opposite relative ordering $k \gg j \gg i$ would violate Condition (8.4). □

Claim 8.4.6. *If there are no Nearly-Identical Alternative Subsets and there exists an ordering $\gg \in L_P$ $|L_P| = 2$*

Proof. This follows immediately from the previous result. If no NIA Subsets exist, Algorithm 14 terminates with output $\gg \in L_P$, and whenever a relative ordering of two alternatives is fixed, the opposite ordering would violate Condition (8.4). However, if $\gg \in L_P$, it can be easily checked that the reverse order $\ll \in L_P$, thus $L_P = \{\gg, \ll\}$. □

8.4.1 Nearly-Identical Alternative Subsets

We have now shown how to identify the ordering $\gg$, for which the data satisfy a mixture model of single-peaked preferences, under the assumption that there are no NIA subsets. In this subsection, we will show how to handle such subsets.

As a starting point, we will again look at Algorithm 14. If NIA subsets are present, at some point it will be impossible to find a subset $|A_k| > 1$ in line (7) to split up in the loop (6-17). Without loss of generality, let us assume there is a single subset $|A'| > 1$, then there is a partial ordering of the alternatives $a_1 \gg \ldots \gg a^- \gg A' \gg a^+ \gg \ldots \gg a_n$. It can be easily proven, by a similar argument as for claim 8.4.5, that for any triple $i, j, k \in A \backslash A'$ and any triple $i, j \in A \backslash A'$, $k \in A'$, Condition (8.4) is satisfied. What remains to be shown is that we can extend $\gg$, such that Condition (8.4)is satisfied for any triple $i \in A \backslash A'$, $j, k \in A'$ and any triple $i, j, k \in A'$.

Consider a pair $(i, j) \in A' \times A'$, such that $\max_{r,s \in A'} (p_{rs}) = p_{ij}$. There are now 3 cases we can distinguish.

1. If $p_{ij} > p_{ia^-}$ and $p_{ij} > p_{ia^+}$, it is clear that no $\gg \in L_P$ exists, as both $a^- \gg j \gg i \gg a^+$ and $a^- \gg i \gg j \gg a^+$ violate Condition (8.4).

2. If (without loss of generality) $p_{ij} > p_{ia^-}$, but $p_{ij} \leq p_{ia^+}$, it must be the case that $a^- \gg i \gg j$. Furthermore, there is no $k \in A'$, $\max_{r,s \in A'} (p_{rs}) \neq p_{ik}$ such that $a^- \gg i \gg j \gg k$, as this would also violate Condition (8.4). As a result, for all $j \in A'$, for which there exists $i \in A'$, $\max_{r,s \in A'} (p_{rs}) = p_{ij}$ and all $k \in A'$ for which there does not exist $i \in A'$, $\max_{r,s \in A'} (p_{rs}) = p_{ik}$, it must be the case that $k \gg j$. At this point, the subset A' is split into two subsets, and Algorithm 14 can be resumed.

3. If both $p_{ij} \leq p_{ia^-}$ and $p_{ij} \leq p_{ia^+}$, there can be no violation of Condition (8.4) for any triple $r \in A \backslash A'$, $s, t \in A'$. Now, the question is whether there exists an ordering $\gg' \in L_{P'}$. This question can again be answered using Algorithm 14. If such an order is found, then both $\gg'$ and its reverse $\ll'$ can be used to complete the partial order $\gg$, in other words, the ordering $a_1 \gg \ldots \gg a'_1 \gg' \ldots \gg' a'_n \gg \ldots \gg a_n$ and $a_1 \gg \ldots \gg a'_n \gg' \ldots \gg' a'_1 \gg \ldots \gg a_n$ are in L_P.

We finish this chapter with the following Theorem.

Theorem 8.4.1. *If $L_P \neq \varnothing$, then an ordering $\gg \in L_P$ can be found in polynomial time.*

Proof. We will structure this proof as follows. First, we show that the theorem holds if there are no NIA subsets. Next, we show that it also holds in the case where such subsets do exist.

First, notice that by combining Claim 8.4.2 and Claim 8.4.3, we show that we can identify an extreme alternative of the ordering $\gg$ using condition (8.7). It is easy to see that given this condition, the extreme alternative can be found by checking all p_{ij} values, of which there are only $O(n^2)$. Next, Claim 8.4.5 then shows that given this extreme alternative, the ordering $\gg$ can be found by using Algorithm 14. We will show that this algorithm runs in polynomial time. The main part of this algorithm is the loop (6-17). In each iteration of this loop, the ordering $\gg$ is refined by splitting up one subset A_k into two or more subsets. Since the loop terminates if there are n such subsets, there are at most $O(n)$ iterations for this loop. Within this loop, step 7 may take $O(n^3)$ steps, since for each triple of alternatives i, j, j' a comparison of p_{ij} and $p_{ij'}$ can be done. This gives a total worst-case bound on the loop and the algorithm as a whole of $O(n^4)$.

Let us now look at NIA subsets. In a given iteration of a loop, it may be the case that no subset A_k can be split up using Algorithm 14. In this case, we pick one subset A' and we must figure out which of the 3 situations distinguished before is relevant. This requires finding the maximum p_{ij}

with $i, j \in A'$, which can be done in $O(n^2)$ steps and as such does not impact the total complexity of the loop (since this is already bounded by $O(n^3)$. Case 1 terminates the algorithm. Case 2 splits the subset in $O(n^2)$ time and we can return to the main algorithm, also without affecting the overall complexity. Finally, we consider Case 3 and suppose $\|A'\| = z$. Notice that this implies Algorithm 14 terminated after $n - z$ iterations of the main loop. If Case 3 occurs, an ordering over the subset of alternatives A' must be found. First, an extreme alternative of A' is identified (or if no such alternative exists, any ordering over A' is permissible) in $O(n^2)$ time. Notice that this means the set A' is split into two subsets. Given this extreme alternative, Algorithm 14 can be run on A', taking $O(z^4)$ time. The complete computation time is then bounded by $O((n-z)n^3 + z^4) = O(n^4)$ and an ordering $\gg \in L_P$ is identified in polynomial time. □

8.5 Conclusion

In this chapter, we presented a mixture model from the choice behaviour literature and applied it to a well-known choice domain from the social choice literature. Necessary and sufficient conditions are derived for the mixture model to hold for single-peaked preferences and a given ordering of the alternatives. Furthermore, we showed that these conditions are easy to check in polynomial time, in contrast to the mixture model for general preferences. Furthermore, a polynomial time algorithm is provided to identify whether or not there exists some ordering of the alternatives for which the mixture model is satisfied.

List of Figures

List of Tables

Bibliography

Afriat, S., 1967a. The construction of separable utility functions from expenditure data, unpublished, Purdue University.

Afriat, S., 1967b. The construction of utility functions from expenditure data. International Economic Review 8 (1), 67–77.

Afriat, S., 1972. The theory of international comparisons of real income and prices. In: International Comparisons of Prices and output. NBER, pp. 11–84.

Afriat, S., 1973. On a system of inequalities in demand analysis: an extension of the classical method. International Economic Review 14 (2), 460–472.

Ahuja, R., Magnanti, T., Orlin, J., 1993. Network Flows: Theory, Algorithms, and Applications. Prentice-Hall.

Alcantud, J., Matos, D., Palmero, C., 2010. Goodness-of-fit in optimizing a consumer model. Mathematical and Computer Modelling 52 (7-8), 1088–1094.

Andreoni, J., Gillen, B., Harbaugh, W. T., 2013. The power of revealed preference tests: ex-post evaluation of experimental design, working Paper.

Andreoni, J., Miller, J., 2002. Giving according to GARP: An experimental test of the consistency of preferences for altruism. Econometrica 70 (2), 737–753.

Apesteguia, J., Ballester, M., 2014. A measure of rationality and welfare, Forthcoming, Journal of Political Economy.

Ballester, M., Haeringer, G., 2011. A characterization of the single-peaked domain. Social Choice and Welfare 36 (2), 305–322.

Bandyopadhyay, T., Dasgupta, I., Pattanaik, P. K., 1999. Stochastic revealed preference and the theory of demand. Journal of Economic Theory 84 (1), 95–110.

Barnett, W., Choi, S., 1989. A Monte Carlo study of tests of blockwise weak separability. Journal of Business & Economic Statistics 7 (3), 363–377.

Bartholdi III, J., Trick, M., 1986. Stable matching with preferences derived from a psychological model. Operations Research Letters 5 (4), 165–169.

Bazaraa, M. S., Jarvis, J. J., Sherali, H. D., 2011. Linear programming and network flows. John Wiley & Sons.

Beatty, T. K., Crawford, I. A., 2011. How demanding is the revealed preference approach to demand? The American Economic Review 101 (6), 2782–2795.

Ben-Or, M., 1983. Lower bounds for algebraic computation trees. In: Proceedings of 15^{th} ACM Symposium on Theory of Computing. pp. 80–86.

Black, D., 1948. On the rationale of group decision-making. The Journal of Political Economy 56 (1), 23.

Block, H., Marschak, J., 1960. Random orderings and stochastic theories of responses. Contributions to probability and statistics 2, 97–132.

Bronars, S. G., 1987. The power of nonparametric tests of preference maximization. Econometrica: Journal of the Econometric Society 55, 693–698.

Burghart, D. R., Glimcher, P. W., Lazzaro, S. C., 2013. An expected utility maximizer walks into a bar... Journal of Risk and Uncertainty 46, 215–246.

Cavagnaro, D. R., Davis-Stober, C. P., 2014. Transitive in our preferences, but transitive in different ways: An analysis of choice variability. Decision 1 (2), 102.

Cherchye, L., De Rock, B., Sabbe, J., Vermeulen, F., 2008. Nonparametric tests of collectively rational consumption behavior: an integer programming procedure. Journal of Econometrics 147 (2), 258–265.

Cherchye, L., De Rock, B., Vermeulen, F., 2007. The collective model of household consumption: a nonparametric characterization. Econometrica 75 (2), 553–574.

Cherchye, L., De Rock, B., Vermeulen, F., 2009. Opening the black box of intrahousehold decision making: theory and nonparametric empirical tests of general collective consumption models. Journal of Political Economy 117 (6), 1074–1104.

Cherchye, L., De Rock, B., Vermeulen, F., 2010. An afriat theorem for the collective model of household consumption. Journal of Economic Theory 145 (3), 1142–1163.

Cherchye, L., De Rock, B., Vermeulen, F., 2011. The revealed preference approach to collective consumption behaviour: Testing and sharing rule recovery. The Review of Economic Studies 78 (1), 176–198.

Cherchye, L., De Rock, B., Vermeulen, F., 2012. Collective Household Consumption Behavior: Revealed Preference Analysis. Vol. 4 of Foundations and Trends in Econometrics. Now.

Cherchye, L., Demuynck, T., De Rock, B., 2014a. Revealed preference analysis for convex rationalizations on nonlinear budget sets. Journal of Economic Theory 152, 224–236.

Cherchye, L., Demuynck, T., De Rock, B.and Hjertstrand, P., 2014b. Revealed preference tests for weak separability: an integer programming approach, forthcoming, Journal of Econometrics.

Cherchye, L., Rock, B. D., Platino, V., 2013. Private versus public consumption within groups: testing the nature of goods from aggregate data. Economic Theory 54, 1248–1258.

Chiappori, P., 1988. Rational household labor supply. Econometrica 56 (1), 63–90.

Choi, S., Fisman, R., Gale, D., Kariv, S., 2007. Consistency and heterogeneity of individual behavior under uncertainty. American Economic Review 97 (5), 1921–1938.

Choi, S., Kariv, S., Müller, W., Silverman, D., 2014. Who is (more) rational? American Economic Review 104, 1518–1550.

Chvátal, V., 1983. Linear programming. WH Freeman and Company, New York.

Coppersmith, D., Winograd, S., 1990. Matrix multiplication via arithmetic progressions. Journal of Symbolic Computation 9 (3), 251–280.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., 2001. Introduction to algorithms, Third Edition. MIT press Cambridge.

Cosaert, S., Demuynck, T., 2013. Revealed preference theory for finite choice sets. Economic Theory, 1–32.

Cox, J., 1997. On testing the utility hypothesis. The Economic Journal 107 (443), 1054–1078.

Crawford, I., De Rock, B., 2014. Empirical revealed preference. Annual Review of Economics 6, 503–524.

Crawford, I., Pendakur, K., 2013. How many types are there? The Economic Journal 123 (567), 77–95.

Davis-Stober, C., 2012. A lexicographic semiorder polytope and probabilistic representations of choice. Journal of Mathematical Psychology 56 (2), 86–94.

De Peretti, P., 2005. Testing the significance of the departures from utility maximization. Macroeconomic Dynamics 9 (03), 372–397.

Dean, M., Martin, D., 2010. How rational are your choice data? In: Conference on Revealed Preferences and Partial Identification.

Deb, R., 2010. An efficient nonparametric test of the collective household model, unpublished, University of Toronto.

Deb, R., Pai, M., 2013. The geometry of revealed preference. Journal of Mathematical Economics 50, 203–207.

Diewert, W., 1973. Afriat and revealed preference theory. The Review of Economic Studies 40 (3), 419–425.

Diewert, W. E., Parkan, C., 1985. Tests for the consistency of consumer data. Journal of Econometrics 30 (1), 127–147.

Dobell, A., 1965. A comment on ayc koo's "an empirical test of revealed preference theory". Econometrica: Journal of the Econometric Society 33 (2), 451–455.

Doignon, J.-P., Falmagne, J.-C., 1994. A polynomial time algorithm for unidimensional unfolding representations. Journal of Algorithms 16 (2), 218–233.

Dridi, T., 1980. Sur les distributions binaires associées à des distributions ordinales. Mathématiques et Sciences humaines 69, 15–31.

Echenique, F., 2014. Testing for separability is hard, working paper, California Institute of Technology.

Echenique, F., Lee, S., Shum, M., 2011. The money pump as a measure of revealed preference violations. Journal of Political Economy 119 (6), 1201–1223.

Elkind, E., Faliszewski, P., 2014. Recognizing 1-euclidean preferences: An alternative approach. In: SAGT.

Escoffier, B., Lang, J., Öztürk, M., 2008. Single-peaked consistency and its complexity. In: ECAI. Vol. 8. pp. 366–370.

Février, P., Visser, M., 2004. A study of consumer behavior using laboratory data. Experimental Economics 7 (1), 93–114.

Fiorini, S., 2006. 0, 1/2-cuts and the linear ordering problem: Surfaces that define facets. SIAM Journal on Discrete Mathematics 20 (4), 893–912.

Fleissig, A. R., Whitney, G. A., 2003. A new pc-based test for varian's weak separability conditions. Journal of Business & Economic Statistics 21 (1), 133–144.

Fleissig, A. R., Whitney, G. A., 2005. Testing for the significance of violations of afriat's inequalities. Journal of Business & Economic Statistics 23 (3), 355–362.

Fleissig, A. R., Whitney, G. A., 2007. Testing additive separability. Economics Letters 96 (2), 215–220.

Fleissig, A. R., Whitney, G. A., 2008. A nonparametric test of weak separability and consumer preferences. Journal of Econometrics 147 (2), 275–281.

Forges, F., Minelli, E., 2009. Afriat's theorem for general budget sets. Journal of Economic Theory 144 (1), 135–145.

Fujishige, S., Yang, Z., 2012. On revealed preference and indivisibilities. Modern Economy 3 (6), 752–758.

Gall, F. L., 2014. Powers of tensors and fast matrix multiplication. In: Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation. pp. 296–303.

Garey, M., Johnson, D., 1979. Computers and intractability. Freeman San Francisco, CA.

Grötschel, M., Lovász, L., Schrijver, A., 1981. The ellipsoid method and its consequences in combinatorial optimization. Combinatorica 1 (2), 169–197.

Grötschel, M., Lovász, L., Schrijver, A., 1993. Geometric Algorithms and Combinatorial Optimization. Springer-Verlag.

Harbaugh, W., Krause, K., Berry, T., 2001. GARP for kids: On the development of rational choice behavior. The American Economic Review 91 (5), 1539–1545.

Håstad, J., 1999. Clique is hard to approximate within 1- ε. Acta Mathematica 182 (1), 105–142.

Heufer, J., 2011. Stochastic revealed preference and rationalizability. Theory and decision 71 (4), 575–592.

Heufer, J., 2012. Testing for utility maximization with error and the loss of power. German Economic Review 13 (2), 161–173.

Houthakker, H., 1950. Revealed preference and the utility function. Economica 17 (66), 159–174.

Houtman, M., Maks, J., 1985. Determining all maximal data subsets consistent with revealed preference. Kwantitatieve methoden 19, 89–104.

Karp, R., 1972. Reducibility among combinatorial problems. Complexity of Computer Computations 40 (4), 85–103.

Karp, R., 1978. A characterization of the minimum cycle mean in a digraph. Discrete mathematics 23 (3), 309–311.

Kawaguchi, K., 2014. Testing rationality without restricting heterogeneity, discussion paper, London School of Economics.

Kitamura, Y., Stoye, J., 2014. Nonparametric analysis of random utility models: Testing, working paper, Cornell University.

Klugkist, I., Hoijtink, H., 2007. The bayes factor for inequality and about equality constrained models. Computational Statistics & Data Analysis 51 (12), 6367–6379.

Knoblauch, V., 2010. Recognizing one-dimensional euclidean preference profiles. Journal of Mathematical Economics 46 (1), 1–5.

Koo, A., 1963. An empirical test of revealed preference theory. Econometrica 31 (4), 646–664.

Koo, A., 1971. Revealed preference–a structural analysis. Econometrica 39 (1), 89–97.

Koo, A., Hasenkamp, G., 1972. Structure of revealed preference: some preliminary evidence. The Journal of Political Economy 80 (4), 724–744.

Laguna, M., Marti, R., Campos, V., 1999. Intensification and diversification with elite tabu search solutions for the linear ordering problem. Computers & Operations Research 26 (12), 1217–1230.

Landsburg, S., 1981. Taste change in the united kingdom, 1900-1955. The Journal of Political Economy 89 (1), 92–104.

Little, I. M., 1949. A reformulation of the theory of consumer's behaviour. Oxford Economic Papers 1 (1), 90–99.

Lübbecke, M. E., Desrosiers, J., 2005. Selected topics in column generation. Operations Research 53 (6), 1007–1023.

Manser, M., McDonald, R., 1988. An analysis of substitution bias in measuring inflation, 1959-85. Econometrica 56 (4), 909–930.

Martí, R., Reinelt, G., 2011. The Linear Ordering Problem: Exact and Heuristic Methods in Combinatorial Optimization. Vol. 175 of Applied Mathematical Sciences. Springer-Verlag Berlin Heidelberg.

Mattei, A., 2000. Full-scale real tests of consumer behavior using experimental data. Journal of Economic Behavior & Organization 43 (4), 487–497.

McFadden, D., 2005. Revealed stochastic preference: a synthesis. Economic Theory 26 (2), 245–264.

McFadden, D., Richter, M., 1990. Stochastic rationality and revealed stochastic preference. In: Preferences, uncertainty, and optimality, essays in honor of Leo Hurwicz. Westview Press, pp. 161–186.

Megiddo, N., 1977. Mixtures of order matrices and generalized order matrices. Discrete Mathematics 19 (2), 177–181.

Megiddo, N., 1979. Combinatorial optimization with rational objective functions. Mathematics of Operations Research 4 (4), 414–424.

Mossin, A., 1972. A mean demand function and individual demand functions confronted with the weak and the strong axioms of revealed preference: an empirical test. Econometrica 40 (1), 177–192.

Munro, I., 1971. Efficient determination of the transitive closure of a directed graph. Information Processing Letters 1 (2), 56–58.

Peters, H., Wakker, P., 1994. WARP does not imply SARP for more than two commodities. Journal of Economic Theory 62 (1), 152–160.

Polisson, M., Quah, J. K., 2013. Revealed preference in a discrete consumption space. American Economic Journal: Microeconomics 5 (1), 28–34.

Préa, P., Fortin, D., 2014. An optimal algorithm to recognize robinsonian dissimilarities. Journal of Classification 31 (3), 351–385.

Quah, J., 2012. A revealed preference test for weakly separable preferences, discussion Paper Series, No. 708, University of Oxford.

Regenwetter, M., Dana, J., Davis-Stober, C., 2010. Testing transitivity of preferences on two-alternative forced choice data. Frontiers in psychology 1 (148), 1–15.

Regenwetter, M., Dana, J., Davis-Stober, C., 2011. Transitivity of preferences. Psychological Review 118 (1), 42.

Regenwetter, M., Davis-Stober, C., 2012. Behavioral variability of choices versus structural inconsistency of preferences. Psychological review 119 (2), 408.

Robinson, W. S., 1951. A method for chronologically ordering archaeological deposits. American Antiquity 16 (4), 293–301.

Rose, H., 1958. Consistency of preference: The two-commodity case. Review of Economic Studies 25, 124–125.

Samuelson, P., 1938. A note on the pure theory of consumer's behaviour. Economica 5 (17), 61–71.

Samuelson, P. A., 1948. Consumption theory in terms of revealed preference. Economica 15, 243–253.

Sippel, R., 1997. An experiment on the pure theory of consumer's behaviour. The Economic Journal 107, 1431–1444.

Smeulders, B., Cherchye, L., De Rock, B., Spieksma, F., 2013. The money pump as a measure of revealed preference violations: A comment. Journal of Political Economy 121 (6), 1248–1258.

Smeulders, B., Cherchye, L., De Rock, B., Spieksma, F., Talla Nobibon, F., 2014a. Transitive preferences in multi-member households, forthcoming, Economic Theory Bulletin.

Smeulders, B., Spieksma, F., Cherchye, L., De Rock, B., 2014b. Goodness of fit measures for revealed preference tests: complexity results and algorithms. ACM Transactions on Economics and Computation 2 (1), Article 3.

Snyder, S., 2000. Nonparametric testable restrictions of household behavior. Southern Economic Journal 67 (1), 171–185.

Suck, R., 1992. Geometric and combinatorial properties of the polytope of binary choice probabilities. Mathematical Social Sciences 23 (1), 81–102.

Swofford, J., Whitney, G., 1994. A revealed preference test for weakly separable utility maximization with incomplete adjustment. Journal of Econometrics 60 (1), 235–249.

Talla Nobibon, F., Cherchye, L., Crama, Y., Demuynck, T., De Rock, B., Spieksma, F., 2013. Revealed preference tests of collectively rational consumption behavior: formulations and algorithms. Research report, KU Leuven, Belgium.

Talla Nobibon, F., Cherchye, L., De Rock, B., Sabbe, J., Spieksma, F., 2011. Heuristics for deciding collectively rational consumption behavior. Computational Economics 38 (2), 173–204.

Talla Nobibon, F., Smeulders, B., Spieksma, F., 2014. An $O(n^2)$ algorithm for testing the generalized axiom of revealed preference, forthcoming, Journal of Optimization Theory and Applications.

Talla Nobibon, F., Spieksma, F., 2010. On the complexity of testing the collective axiom of revealed preference. Mathematical Social Sciences 60 (2), 123–136.

Tarjan, R., 1972. Depth-first search and linear graph algorithms. SIAM J. Comput. 1, 146–160.

Trick, M., 1989. Recognizing single-peaked preferences on a tree. Mathematical Social Sciences 17 (3), 329–334.

Tsur, Y., 1989. On testing for revealed preference conditions. Economics Letters 31 (4), 359–362.

Tutte, W. T., 1961. On the problem of decomposing a graph into n connected factors. Journal of the London Mathematical Society 1 (1), 221–230.

Valiant, L. G., 1979. The complexity of enumeration and reliability problems. SIAM Journal on Computing 8 (3), 410–421.

Varian, H., 1982. The nonparametric approach to demand analysis. Econometrica 50 (4), 945–973.

Varian, H., 1983. Non-parametric tests of consumer behaviour. Review of Economic Studies 50 (1), 99–110.

Varian, H., 1990. Goodness-of-fit in optimizing models. Journal of Econometrics 46 (1-2), 125–140.

Varian, H., 1993. Goodness-of-fit for revealed preference tests, unpublished.

Varian, H., 2006. Revealed preference. In: Szenberg, M., Ramrattan, L., Gottesman, A. (Eds.), Samuelsonian economics and the twenty-first century. Oxford University Press, pp. 99–116.

Varian, H. R., 1985. Non-parametric analysis of optimizing behavior with measurement error. Journal of Econometrics 30 (1), 445–458.

Vermeulen, F., 2002. Collective household models: principles and main results. Journal of Economic Surveys 16 (4), 533–564.

Warshall, S., 1962. A theorem on Boolean matrices. Journal of the ACM 9 (1), 11–12.

Williams, V. V., 2012. Multiplying matrices faster than coppersmith-winograd. In: Proceedings of the forty-fourth annual ACM symposium on Theory of computing. ACM, pp. 887–898.

Yao, A.-C., 1991. Lower bounds for algebraic computation trees with integer inputs. SIAM Journal on Computing 20 (4), 655–668.

Zuckerman, D., 2006. Linear degree extractors and the inapproximability of max clique and chromatic number. In: Proceedings of the thirty-eighth annual ACM Symposium on Theory of Computing. ACM, pp. 681–690.

Doctoral dissertations from the Faculty of Business and Economics

A list of doctoral dissertations from the Faculty of Business and Economics can be found at the following website:
`http://www.kuleuven.ac.be/doctoraatsverdediging/archief.htm`.