# Measuring Rationality with the Minimum Cost of Revealed Preference Violations

Mark Dean and Daniel Martin

# Online Appendices - Not for Publication

# 1 Algorithm for Solving the MASP

In this online appendix we formally define the maximal acyclical set problem (MASP) and describe our algorithm for solving the problem. A MASP can be solved by identifying and solving a related minimum set covering problem (MSCP) – a class of problems which is well studied within the field of operations research. This is both good news and bad news for solving incidences of MASP. The bad news is that MSCP (and so MASP) are *NP-hard*. The implication is that there is no known method that can guarantee solution times will increase only as a polynomial function of the number of inputs to MASP. In other words, one cannot guarantee that solution times will not get very large, very quickly as the size of the input data grows. However, the good news is that researchers in operations research have developed a number of tools that *in practice* solve MSCP quickly and exactly for many data sets. The equivalence between MSCP and MASP means that these techniques can be adopted wholesale for solving MASP.

Solving MASP is equivalent to calculating the Houtman-Max Index (HMI). To adjust the algorithm to calculate Minimum Cost Index (MCI), all that is required is to (1) remove relations rather than observations, (2) minimize cost instead of size, and (3) divide the solution by total expenditure.

## 1.1 Definitions

MASP is the problem of finding the size of the largest subset of a set of choice data that generates acyclical revealed preference relations. The primitives of the problem are a grand set of *alternatives* $Z$, a set of *observations* $X$ and a *relation function* $D : X \rightarrow 2^{Z \times Z}$ that characterizes a set of binary relations on $Z$ generated by each observation in $X$. We call the triple $\{Z, X, D\}$ a *data set*.[1]

[1] We assume that $X$ is finite. In this case, we can solve MASP whatever the cardinality of $Z$. Moreover, acyclicality is enough to guarantee that choices can be rationalized by utility maximization even if $Z$ is uncountable. This is because we can concentrate on the (finite) set of objects $\overline{Z}$ that are chosen in any observation $X$. If the data is acyclic, we can generate a utility function $u : \overline{Z} \rightarrow \mathbb{R}$ that rationalizes choice

As an example, consider the case of a laboratory experiment in which we observe a subject making choices from subsets of $Z$. Furthermore, assume that we are prepared to say that the chosen object in any set is strictly preferred to all the other available alternatives.[2] In this case we could think of each observation in $X$ as consisting of a tuple $(z, A)$ with $z \in A$ and $A \in 2^Z/\emptyset.$, implying that alternative $z$ has been observed as being chosen from the set of alternatives $A$. The function $D : X \rightarrow 2^{Z\times Z}$ would then be defined as the revealed preference relations generated by $X$:

$$\begin{aligned} \forall\ (z, A) &\in X, \\ D(z, A) &= \{(z, y) \mid y \in A/\{z\}\} \end{aligned}$$

We denote by the binary relation $\succ_x \subset Z \times Z$ the relations generated by the observation $x \in X$, so that $\succ_x = D(x)$. For any $B \subset X$, we define the binary relation $\succ^B$ on $Z$ as

$$z \succ^B w \text{ if, for some } x \in B,\ z \succ_x w.$$

For an arbitrary binary relation $\succ$, a *cycle* refers to a set of alternatives $z_1, z_2, ..., z_n \in Z$ such that $z_1 \succ z_2 \succ .... \succ z_n \succ z_1$. We say that a set of observations $B \subset X$ is *acyclic* if the binary relation $\succ^B$ generated by $B$ contains no cycles. Thus, we define MASP as the problem of finding the size of the largest subset $B \subset X$ such that the resulting binary relation $\succ^B$ is acyclic.

**Definition 1** *The maximal acyclical set problem (MASP) for a data set $\{Z, X, D\}$ is the problem of finding the size of a set $B \subset X$ such that*

*(i) $B$ is acyclic*

*(ii) if $B' \subset X$ and $|B'| > |B|$, then $B'$ is not acyclic*

between these objects. All remaining alternatives can be assumed to have a utility equal to $\min_{z\in\overline{Z}} u(z) - 1$.

[2] Note that this assumption is *not* central to our methodology, which is flexible enough to cope with almost any definition of revealed preference. For example, if we are observing choices from budget sets, we could say that $x$ is revealed preferred to $y$ only if $x$ was chosen when $y$ was available at strictly lower cost. The precise assumption is defined by the nature of the mapping $D$.

In other words, MASP is the problem of finding the size of the largest acyclical subsets of $X$. Note that the maximal acyclical set may not be unique.

Next, we define a MSCP. In order to explain the idea behind this class of problems, we illustrate it with the following example:

**Example 1** *Imagine you are setting up a cellphone network and need to buy rights to bandwidth in all 50 states. However, bandwidth is being sold in packages of different states (e.g. package 1 includes Alabama, Rhode Island and Wyoming, package 2 includes South Dakota, Minnesota and Wyoming and so on). Each package has a particular cost. The problem you face as a cellphone provider is: "What collection of packages should I buy to ensure some bandwidth in all 50 states at the lowest possible cost?" In other words, what is the minimum cost way of covering all 50 states?*

A formal statement of this class of problems is as follows:[3]

**Definition 2** *Let $S$ be a (finite) set , $\Theta \subset 2^S$ be a collection of subsets of $S$ and $k : S \rightarrow \mathbb{R}$ be a cost function which attaches a cost to each element of $S$. A covering of $\Theta$ is a subset $T \subset S$ such that $\theta \cap T \neq \varnothing \; \forall \; \theta \in \Theta$. In other words, every set in $\Theta$ contains at least one element of $T$. A minimum set covering problem (MSCP) is the problem of finding the minimum cost of covering of $\Theta$, or*

$$\min_{T \in 2^S} \sum_{s \in T} k(s)$$

$$\textit{subject to } \theta \cap T \neq \varnothing \; \forall \; \theta \in \Theta$$

Again, note that the minimum covering set may not be unique.

In the bandwidth example above, we can let $S$ be the set of packages and $\Theta$ be a collection of 50 sets, one for each state, containing the packages which cover each state (e.g. if Alabama was covered by packages 1, 7 and 9 then $\theta_1 = \{1, 7, 9\}$, if Alaska was covered by packages 3,

[3] Note that some people call the problem stated in this way as the 'minimum hitting problem'. However, Ausiello et al. [1980] show that this is equivalent to other statements of the minimum set covering problem.

14, 19 and 23 then $\theta_2 = \{3, 14, 19, 23\}$ and so on). $k$ would contain information on the cost of each package.

## 1.2 Equivalence of MASP and MSCP

In order to show the equivalence of MASP and MSCP, we need to formalize the concept of the set of cycles generated by a data set.

**Definition 3** *A* ***cycle*** *generated by a data set $\{X, Z, D\}$ consists of a non-repeating sequence $z_1, ..., z_n$ in $Z$ and a sequence $x_1, ..., x_n$ in $X$ such that*

$$z_1 \succ_{x_1} \ldots \succ_{x_{n-1}} z_n \succ_{x_n} z_1$$

Let $C$ denote the set of all cycles generated by $X$.

We will say that observation $x \in X$ *breaks* a cycle $c \in C$ if $x$ appears in the sequence $x_1, ..., x_n$. Note that if a subset $B$ of $X$ breaks all cycles in $C$, then the complement of that subset $X/B$ is acyclic.

Next, we will define the components of a *complimentary* MSCP for a particular MASP in the following way.

**Definition 4** *For the MASP associated with a data set $\{X, Z, D\}$, we define the complimentary minimum set covering problem by the following elements $\bar{S}$, $\bar{\Theta}$ and $\bar{k}$:*

1. $\bar{S} = X$
2. $\bar{k}(x) = 1 \; \forall \; x \in S$
3. $\bar{\Theta} = \left\{\overline{\theta}(c) \mid c \in C\right\}$*, where* $\overline{\theta}(c) = \{x \in X \mid x \text{ breaks } c\}$

With this structure, the problem of finding the size of the smallest subset of $X$ which breaks all cycles in $C$ is the same as finding the cost of the minimum covering of $\Theta$. Further, a smallest subset of $X$ which breaks all cycles in $C$ is the complement of a largest subset of $X$ that is acyclic.

**Theorem 1** *For the MASP associated with a data set $\{X, Z, D\}$, if some number $s$ is the solution to the complimentary MSCP ($\bar{S}$, $\bar{\Theta}$, $\bar{k}$), then $|X| - s$ is the solution to the MASP.*

**Proof.** See Appendix 3. ■

Thus any MASP can be solved by solving the equivalent MSCP. While the MSCP is NP-hard, these problems have been studied exhaustively in the operations research literature because they can be applied to many real world situations, such as train scheduling and city planning. As a result, algorithms have been developed to solve or approximate solutions to MSCP quickly for larger and larger data sets.

## 1.3 A Description of the Algorithm

Using the result of Theorem 1 to solve MASP associated with a data set $\{X, Z, D\}$ requires two algorithmic components. First, in order to construct the set $\bar{\Theta}$, we need an algorithm that identifies the set of all cycles $C$ generated by MASP, as well as the observations $x \in X$ that break each cycle $c \in C$. Second, we need an algorithm to solve the complimentary MSCP. Here we describe briefly how we implement each of these stages.

In order to find the set of cycles of $C$, we use a modification of Johnson's algorithm (Johnson [1975]) – a computationally efficient graph theoretic algorithm. Johnson's algorithm is based on 'depth-first' search, a standard approach to finding cycles, which looks at the objects preferred to an initial object, then looks for the objects that are preferred to the first of those preferred objects and so on until a cycle is found or the process terminates. At that point, the algorithm goes back one level and proceeds from the second preferred object until all possibilities are exhausted. To gain efficiency, Johnson adds a blocking function to prevent redundant searching on the tree, which gives it a computation time upper bound of $O\left(\left(n + e\right)\left(c + 1\right)\right)$, where $n$ is the number of nodes (in this case $|Z|$), $e$ is the number of edges ($|D(X)|$) and $c$ is the number of cycles ($|C|$).

In order to increase efficiency, we first apply the elementary rules of Guardabassi [1971] to absorb dominated nodes and remove singular edges, which are inconsequential for finding a solution to MASP. Note that this technique cannot be employed when calculating the Minimum Cost Index (MCI).

Next, we need a method for solving the companion MSCP, which is NP-hard. These problems have been studied exhaustively in the operations research literature because they can be applied to many real world situations, such as train scheduling and city planning. As a result, algorithms have been developed to solve or approximate solutions to MSCP quickly for larger and larger data sets.

The most widely used solution methods for MSCP are a class of branch and bound algorithms that find an exact solution by iteratively 'relaxing' the corresponding binary integer programming problem so that linear programming techniques can be used to create bounds on the problem. When MSCP is written as a binary integer programming problem, each cover is given a value of 0 or 1, where 1 signifies that a cover is included in the minimum cover. With this specification, MSCP can be written in the form:

$$\min_{x \in \{0,1\}^n} f' * x \quad \text{subject to } Ax \geq b$$

where $n$ is the number of covers, $x$ is a vector of cover values, $f$ is a vector that includes the cost of each cover, $b$ is a vector of 1's and $a_{ij} = 1$ if cover $j$ covers node $i$. This problem is relaxed by allowing the value assigned to each cover to be any real number between 0 and 1, which gives a lower bound to the solution of the binary integer problem. Now imagine the binary integer problem as a tree, where each fork is a decision whether or not to include a single cover in the minimum cover. The bound given by the relaxed problem allows the algorithm to remove many branches of the tree from consideration.

These algorithms have been integrated into standard Integer Programming (IP) software packages. Many programming languages include optimization functions that use an internal IP solver (including Matlab). There are also a variety of specialist solvers available on a commercial (e.g. CPLEX) and noncommercial (e.g. SCIP and MINTO) basis. For our analysis, we use the GLPK callable C library (www.gnu.org/software/glpk/), which acts as an internal solver for our C++ program. By using an internal solver, we can easily pass data from Johnson's algorithm to the IP solver.

# 2 MASP and NP Completeness

One advantage of connecting MASP with an associated MSCP is that it allows us to relate MASP to the concept of *NP-completeness.* Introduced by Cook [1971], NP-completeness is a property of decision problems. The set of NP-complete problems is a subset of the class of problems that are NP (which stands for Nondeterministic Polynomial time). A problem is NP if its solution can be verified in *polynomial time*[4]; though importantly, a solution cannot necessarily be *found* in polynomial time. A decision problem $M$ is described as NP-complete if it satisfies two properties:

1. $M$ is NP
2. Every problem in NP is reducible[5] to $M$

A problem is called *NP-hard* if it satisfies condition 2, whether or not it satisfies condition 1.

The class of NP-complete problems is interesting because, if any such problem can be solved in polynomial time (denoted as the class P of decision problems), then any problem in NP can be solved in polynomial time. In other words, this would imply that any problem whose solution can be verified in polynomial time can also be solved in polynomial time. Whether or not this is true is one of the big open problems in mathematics.[6] Thus, the quest to find an algorithm that will solve an NP-complete problem in polynomial time is taking on one of the more difficult problems of mathematics.

MSCP as we have described it above is NP-hard rather than NP-complete, because a solution cannot be verified in polynomial time. The *decision* version of MSCP, which asks whether there is a cover of $\Theta$ with total cost less than some value $V$, is NP-complete (Karp

[4] The run time of the verifier is no greater than a polynomial of the problem size.

[5] Meaning there is a (polynomial time) transformation of any problem in NP to $M$. Thus, $M$ can be used once as a subroutine to solve any problem in NP.

[6] The Clay Mathematics Institute is offering a prize of $1 million for anyone who can prove either P=NP or P≠NP.

[1972]). However, the property of NP-hardness alone means that finding a polynomial time solution algorithm for the *optimization* version of MSCP would be enough to show that all NP-complete problems could be solved in polynomial time, making such a algorithm beyond the current state of mathematical knowledge.

Does this mean that there is no known way of solving MASP in polynomial time? To prove this, we have to show that MASP is NP-hard. The standard way of doing this is to show that a known NP-complete problem is reducible to MASP, which implies that *all* NP problems are reducible to MASP. We show this to be the case by showing that the decision version of MSCP is reducible to MASP. This implies that the search for an algorithm that is guaranteed to solve MASP in polynomial time is essentially futile.

# 3 Proofs

**Theorem:** For the MASP associated with a data set $\{X, Z, D\}$, if some number $s$ is the solution to the complimentary MSCP ($\bar{S}$, $\bar{\Theta}$, $\bar{k}$), then $|X| - s$ is the solution to the MASP.

**Proof.** Let $T$ be a minimum cost covering of $\bar{S}$. In order to prove the result, we need to show two things about the set $A = X/T$: First, that $A$ itself is acyclic. Second, that any set $A' \subset X$ such that $|A'| > |A|$ contains a cycle. We prove both of these claims by contradiction.

1. Assume $A$ is not acyclic. Then, there exists some cycle $z_1 \succ^A ... \succ^A z_n \succ^A z_1$. But this implies there exists some cycle $c \in C$ such that $\overline{\theta}(c) \subset A$, contradicting the claim that $T \cap \overline{\theta} \neq \varnothing$ for all $\overline{\theta}$.

2. Let $A'$ be an acyclic set such that $|A'| > |A|$, and consider the set $T' = X/A'$. It must be true that $T^{'} \cap \overline{\theta} \neq \varnothing \; \forall \; \overline{\theta} \in \bar{\Theta}$. If not, then there exists some $\overline{\theta} \in \bar{\Theta}$ such that $\overline{\theta} \subset A'$. This in turn implies that

$$\exists \; z_1 \succ_{x_1} \ldots \succ_{x_{n-1}} z_n \succ_{x_n} z_1$$

$$\Rightarrow z_1 \succ^{A'} ... \succ^{A'} z_n \succ^{A'} z_1$$

and so $A'$ would not be acyclic. But, as $|A'| > |A|$, then $|T| > |T'|$, in turn implying that

$$\sum_{s \in T} \overline{k}(s) = |T| > |T'| = \sum_{s \in T'} \overline{k}(s)$$

which contradicts the fact that $T$ is a minimum cost covering of $\bar{S}$.

■

**Theorem:** MASP is NP-hard.

**Proof.** To show that MASP is NP-hard, we need to show that a known NP-complete problem can be reduced to MASP. We will use the decision version of MSCP, which was shown to be NP-complete by Garey and Johnson [1979]: Given a (finite) set $S$, a collection of subsets $\Theta \subset 2^S$, and an integer $k$ does there exist a covering $T$ of $\Theta$ of size $k$ or less?

Let $X = S$ and $Z = S$, and apply an arbitrary index $I$ to the elements of $S$. Next, define a function

$$\begin{aligned} c(\theta) &= \{(s_i, s_j) \mid s_i, s_j \in \theta \text{ and } i < j \text{ and } \nexists k \text{ s.t. } i < k < j\} \\ &\cup \{(s_j, s_i) \mid s_i, s_j \in \theta \text{ and } \nexists k \text{ s.t. } k < i \text{ and } j > k\}. \end{aligned}$$

This creates a "cycle" from each cover that begins with the lowest indexed object, moves to the highest indexed object, and ends with the lowest indexed object again. For example, $c(\{s_2, s_3, s_5\}) = \{(s_2, s_3), (s_3, s_5), (s_5, s_2)\}$.

Finally, let $R = \cup_{\theta \in \Theta} c(\theta)$ and

$$D(s_i) = \{r \in R \mid r_1 = s_i\}.$$

Clearly, this simple structure $\{X, Z, D\}$ can be added in polynomial time. Further, this structure can be used to formulate a MASP, which returns $M$, the size of a maximal acyclical set $A$.

We claim that $|X| - |A| \leq |T|$ if and only if there exists a covering of $\Theta$ of size $k$ or less.

First, assume that $|X| - |A| \leq |T|$ and that there does not exist a covering of $\Theta$ of size $k$ or less. Let $T^* = X/A$. It must be that $T^*$ covers all $\theta \in \Theta$ because $T^*$ breaks the cycles in $D$ by including a member of each cycle, and thus, it contains a member of each $\theta \in \Theta$. But then $T^*$ is a covering of size $k$ or less, contradicting the latter assumption.

Second, assume that there exists a covering of $\Theta$ of size $k$ or less and that $|X| - |A| > |T|$. Let $A^* = X/T$. It must be that $A^*$ is acyclic because $T$ contains an element from every cycle created by the $c$ function. But then by the second assumption

$$\begin{aligned} |X| - |A| &> |X| - |A^*| \\ |A| &< |A^*| \end{aligned}$$

which contradicts that $A$ is a maximal acyclical set. ■