Erc Sets and Antimatroids[1]

Nazarré Merchant and Jason Riggle
Eckerd College and the University of Chicago
August 31, 2012

Abstract
Grammars in Optimality Theory can be characterized by sets of Ercs (Elementary Ranking Conditions). Antimatroids are structures that arose initially in the study of lattices. In this paper we prove that antimatroids and consistent Erc sets have the same formal structures. We do so by defining two functions *MChain* and *RCerc*, *MChain* being a function from consistent sets of Ercs to antimatroids and *RCerc* a function from antimatroids to Erc sets. We then show that these functions are inverses of each other and that both maintain the structural properties of Erc sets and antimatroids. This establishes that antimatroids and consistent Erc sets have the same formal structure, allowing linguists to import from the sizable work done on antimatroids any and all results.

Keywords: Optimality Theory, Ercs, Antimatroids

1 Introduction

In this paper we show that Optimality Theory (OT), characterized by sets of Ercs (Elementary Ranking Conditions) is identical to the theory of antimatroids (Dilworth 1940, see Korte et al. 1991 or Monjardet 1985 for an extended discussion on their independent discovery by other researchers) sharing all formal properties with this theory (this equivalence was first observed informally in Author 2010 and was independently discovered by Prince (see Prince forthcoming)). Because the two formal systems are the same, any result that holds for antimatroids holds for an optimality theoretic grammar, and vice versa. We hope that this equality will prove fruitful for linguists, as there is a large body of work on antimatroids, all of which is immediately portable to an optimality theoretic framework.

Antimatroids, initially discovered by Dilworth while studying lattices, have arisen in many other fields including modeling scheduling problems, task planning, and modeling knowledge states of human learners (Monjardet 1985 covers much history here). There are several different, but equivalent, definitions of antimatroids; the definition that we will be using here is that an antimatroid is an *accessible set system closed under union*. These terms will be defined later in this paper, but at its roughest, an antimatroid is a type of subset of a power set. For our purposes we will consider the power set over a set of OT constraints. This will provide for us an easy way to map OT grammars to antimatroids.

---

[1] Much thanks goes to Alan Prince for regular discussions about this paper, discussions which greatly facilitated its completion. The provenance of errors and miscommunications in the paper falls at the feet of its two authors.

The formal characterization of optimality theory that we will be comparing antimatroids to is the one presented in Prince (2002), in which the ranking requirements of a given language are represented using Elementary Ranking Conditions (Ercs). Ercs are vectors of length n, where n is the number of constraints in the system, composed of the symbols L, e, and W. An Erc encodes logical statements about rankings. To recapitulate Prince 2002 briefly, the meaning of an Erc is that at least one constraint whose corresponding coordinate contains a W outranks all of the constraints whose coordinates contain Ls. Thus, given four constraints (a, b, c, d), the Erc ⟨W, e, L, L⟩ means that constraint a outranks both constraint c and constraint d and is mute on the ranking of constraint b, while the Erc ⟨W, W, L, L⟩ encodes the fact that either a or b outranks both c and d.

The set of constraints marked with W in an Erc α will be denoted $W(\alpha)$, and likewise the set of constraints marked with L will be denoted $L(\alpha)$. Using this notation, the Erc α expresses the condition that at least one constraint in $W(\alpha)$ outranks all of the constraints in $L(\alpha)$. Rankings that obey the conditions expressed by an Erc are said to *satisfy* the Erc and, more generally, given a set of Ercs E, rankings that satisfy all of the Ercs in E are said to satisfy E.

In this paper we will define two functions, *MChain* and *RCerc*, *MChain* being a function from sets of consistent Ercs to antimatroids and *RCerc* a function from antimatroids to sets of consistent Ercs. The main result of this paper will be to show that these two functions are inverses of one another and that these functions preserve the respective structures of Erc sets and antimatroids under their mapping. This will establish an isomorphism between consistent Erc sets and antimatroids, demonstrating that they are formally the same.

2 Representing Ercs in Lattices

Consistent Erc sets over a set of constraints can be naturally represented in a lattice made from the power set over that constraint set (we call this lattice the power set lattice). In this section we will show how this representation comes about by first demonstrating how a total order consistent with an Erc set can be represented in a power set lattice, and then using this encoding to define a function from Erc sets to power set lattice, which we will call *MChain*. This *MChain* function will be the basis of our function from Erc sets to antimatroids.

2.1 From Erc Sets to Linear Extensions

A set of Ercs delimits a set of rankings and if the ranking-set is not empty, we say that the Erc set is *consistent*. We will call rankings (total orders on the constraints) that satisfy all the members of an Erc set the *linear extensions* of that Erc set.

To see this correspondence in action, consider the set of Ercs E = {⟨W, e, e, L⟩, ⟨W, W, L, L⟩}. What we're interested in is the set of linear extensions that satisfy the both of these Ercs. It turns out that each Erc from E, *individually*, picks out twelve linear extension, but of course not the *same* twelve linear extensions. The overlap, that is the linear extensions that satisfy both Ercs, number nine. Below we will consider the linear extensions that satisfy each of the two Ercs individually, and determine the intersection of these two sets.

Starting with the Erc ⟨W, e, e, L⟩, if we label our four constraints *a*, *b*, *c*, and *d*, this Erc picks out those linear extensions in which $a \gg d$. In Table 1 below all linear extensions over four constraints are listed, numbering 4! = 24; the 12 linear extensions corresponding to ⟨W, e, e, L⟩ are bolded and in blue.

Table 1. List of all linear extensions over four constraints {*a*, *b*, *c*, *d*} (bolded and in blue correspond to ⟨W, e, e, L⟩)

- ***abcd*** ***abdc*** ***acbd*** ***acdb*** ***adbc*** ***adcb***
- ***bacd*** ***badc*** ***bcad*** *bcda* *bdac* *bdab*
- ***cbad*** ***cbda*** ***cabd*** *cadb* *cdba* *cdab*
- *dbca* *dbac* *dcab* *dcab* *dabc* *dacb*

The second Erc, ⟨W, W, L, L⟩, also picks out a set of twelve linear extension, those in which constraint *a* or *b* dominates constraint *c* and *d*. There are twelve linear extensions that satisfy this condition. Those twelve are highlighted in bold below, and are listed amongst the total 24 linear extensions over four constraints listed in Table 2.

Table 2. List of all linear extensions over four constraints {*a*, *b*, *c*, *d*}, bolded and in brown corresponding to ⟨W, W, L, L⟩

- ***abcd*** ***abdc*** ***acbd*** ***acdb*** ***adbc*** ***adcb***
- ***bacd*** ***badc*** ***bcad*** ***bcda*** ***bdac*** ***bdab***
- *cbad* *cbda* *cabd* *cadb* *cdba* *cdab*
- *dbca* *dbac* *dcab* *dcab* *dabc* *dacb*

Of course, given that *both* ⟨W, e, e, L⟩ and ⟨W, W, L, L⟩ hold, the linear extensions that satisfy Erc set E are those that are in the intersection of the two linear extension sets highlighted in Table 1 and Table 2. The intersection contains seven linear extensions and are the bolded linear extensions in the table below.

Table 3. List of all linear extensions over four constraints {*a*, *b*, *c*, *d*}, bolded corresponding to ⟨W, W, L, L⟩ *and* ⟨W, e, e, L⟩

- ***abcd*** ***abdc*** ***acbd*** ***acdb*** ***adbc*** ***adcb***
- ***bacd*** ***badc*** ***bcad*** *bcda* *bdac* *bdab*
- *cbad* *cbda* *cabd* *cadb* *cdba* *cdab*
- *dbca* *dbac* *dcab* *dcab* *dabc* *dacb*

The property that the linear extensions consistent with a set of Ercs is the intersection of the set of linear extensions consistent with each individual Erc is, of course, more general. That is, given, S = {Erc_i}, a set of Ercs and their corresponding linear extensions {LE_i}, where LE_i is the set of linear extensions consistent with Erc_i, then the set of linear extensions consistent with S is the intersection of LE_i. Focusing on linear extensions consistent with an Erc set turns out to be a useful characterization of Erc sets because of the way that linear extensions can be embedded in lattices, as discussed in the next section.

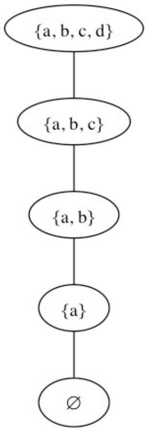
2.2 From Linear Extensions to Lattices

A set of linear extensions consistent with a set of Ercs, while capturing all ranking requirements of that set of Ercs, is often a nearly opaque object, obscuring the interrelations amongst the constraints that may obtain. One way of elucidating these relations is by embedding linear extensions in a lattice, first by reducing a linear extension to a subset of the power set over the constraints in the system, and then from that set building up a lattice.

To see how this is done, first consider the linear extension *abcd* (that is, the total order $a \gg b \gg c \gg d$). We can encode this ordering as a set of sets, called the *maximal chain*, and denoted *MChain*(*abcd*) = {{}, {*a*}, {*a*,*b*}, {*a*,*b*,*c*}, {*a*,*b*,*c*,*d*}}. This set encodes ranking information via the formulation that the set of size n contains the n highest ranked constraints. So in this example, {*a*}, the set of size 1, contains *a*, signifying that *a* is the highest ranked constraint, i.e. *a* is undominated. The set of size 2, {*a*, *b*}, contains the two highest ranked constraints, *a* and *b*. This set {*a*, *b*} only states that *a* and *b* fill the first two ranks of the total order – it is mute on the relative ranking between them. We can deduce though that since the set of size 1 contains *a*, *a* must outrank *b* and so constraint *b* is the second highest ranked constraint. The remaining ranking requirements follow similarly from the set of size 3 and the set of size 4, ensuring that our linear extension *abcd* is exactly encoded in the *MChain*(*abcd*).

The relation of subset inclusion gives these five sets, from *MChain*(*abcd*), the structure of a lattice (albeit a very simple one), one from which ranking relations can be straightforwardly read. So, consider the lattice in Figure 1 below, produced from the five sets contained in *MChain*(*abcd*). To recover the ranking from the lattice, one starts at the empty set and proceeds *up* the lattice, ranking each new constraint as one encounters it. So, moving from the empty set to {*a*}, one ranks the constraint *a* at the top of the hierarchy. Constraint *b* is encountered next in the set {*a*, *b*}, requiring that it be ranked next. With {*a*, *b*, *c*}, *c* is next ranked. The ranking ends with the addition of *d* in the set {*a*, *b*, *c*, *d*} at the top of the lattice.

Figure 1. The lattice associated with *MChain*(*abcd*)



We have just encoded one linear extension into a lattice. In fact, we can encode *all* linear extensions into a *single* lattice, that lattice being the one that contains all sets of different selections of constraints. This is the power set lattice over the set of constraints, ordered by set inclusion. Because the power set lattice contains all subsets of the power set of the constraints as nodes on the lattice, each possible linear extension of a constraint set can be read off in a manner identical to the one laid out above for *MChain*(*abcd*).

Demonstrating this encoding, below in Figure 2 we have the power set lattice for the four constraints $\{a, b, c, d\}$. The *MChain*(*abcd*) is shown in the bolded portion of the lattice. Other rankings are captured similarly. So, for example, the ranking *acdb*, represented as *MChain*(*acdb*) is shown in Figure 3, also as a subset of the power lattice.

Figure 2. The power lattice over $\{a, b, c, d\}$ with *MChain*(*abcd*) in bold

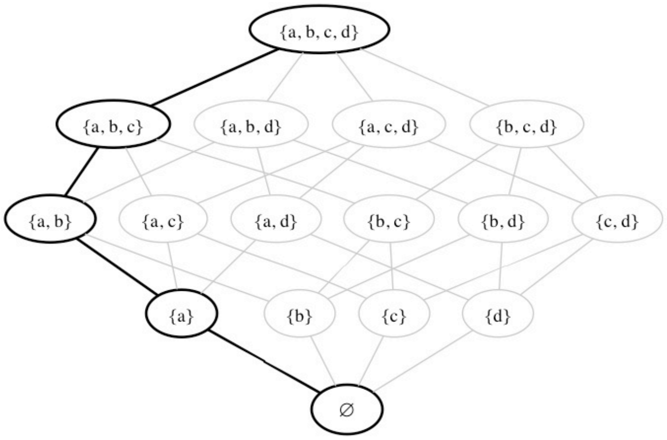
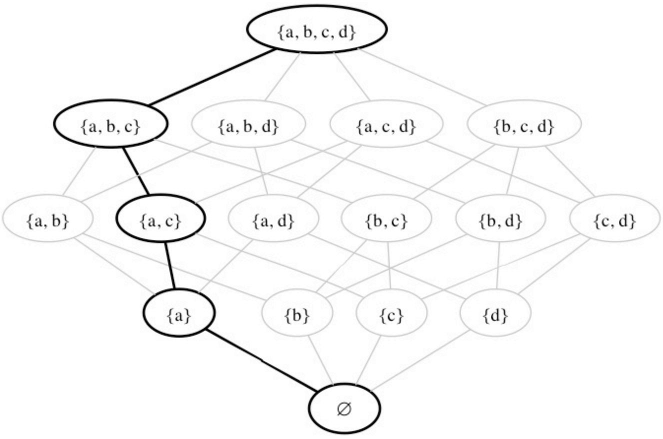


Figure 3. The power lattice over $\{a, b, c, d\}$ with *MChain*(*acbd*) in bold



Note that each linear extension's encoding in the power lattice traces a path through the lattice starting at the empty set at the bottom of the lattice, working its way up to the top of the lattice to the set containing all the constraints. These paths in the power lattice, from the bottom element to the top element are called *maximal chains*, corresponding to our previously defined function *MChain*, and each maximal chain corresponds to a linear extension, and vice versa.

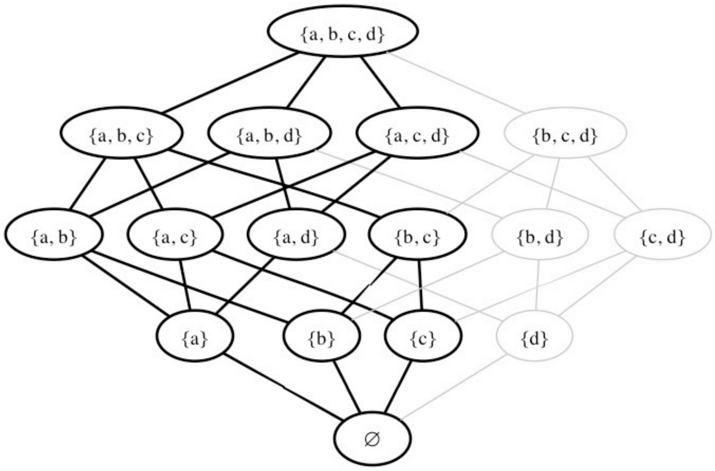
One can readily count that there are 24 maximal chains in the power lattice over four constraints, one for each of the 24 total orderings on the constraint set $\{a, b, c, d\}$. As is evident the maximal chains overlap; so, for example the linear extensions *abcd* and *acbd* share four of their five sets in the power lattice only differing in the relative order of b and c and so only differing in their inclusion of the sets $\{a, b\}$ and $\{a, c\}$.

An immediate consequence of being able to encode any set of linear extensions in the power lattice is that a set of Ercs can be encoded in the power lattice over a constraint set. A set of Ercs, as discussed above, defines a set of linear extensions. Each of these linear extensions resides in the power lattice, and the sub-lattice consisting of their representations exactly represents the set of Ercs they are derived from.

We can see this process by considering the Erc $\langle$W, e, e, L$\rangle$ which requires that $a \gg d$. The lattice encodes the fact that constraint a, b, or c may be topped ranked, and it does so by including in the lattice the nodes $\{a\}$, $\{b\}$, and $\{c\}$. If the topped-ranked constraint is a, any of the remaining constraints can be ranked in any order, since a ranked in the top spot immediately satisfies the condition $a \gg d$. This freedom to rank any remaining constraint once a is top-ranked is captured by the fact that all supersets of $\{a\}$ are included in the lattice.

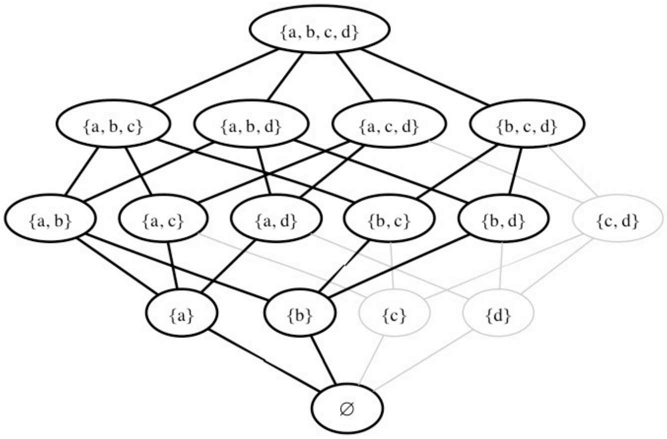
Looking to the remaining possible rankings, we see that the twelve maximal chains that correspond to the twelve linear extentions that satisfy the Erc $\langle$W, e, e, L$\rangle$ are easily ensconsed in a single power lattice over $\{a, b, c, d\}$, shown below in Figure 4. The twelve maximal chains are bolded in the power lattice in Figure 4. We denote the bolded portion by *MChain*($\langle$W, e, e, L$\rangle$). A word of warning: we are being slightly loose with out notation here – previously *MChain*(–) was a function on a single linear extension to a subset of the power set of the constraints, while here it's a function on an Erc α, producing a subset of the power set of constraints that is composed of the union of all *MChain*(λ) where λ runs over all the linear extensions consistent with α. So, for a linear extension λ, *MChain*(λ) corresponds to the maximal chain from order theory, while *MChain*(α) on an Erc α has a looser definition, being composed of unions of maximal chains. We will also extend this looseness of the domain of *MChain*(–) to include Erc sets shortly.

Figure 4. The power lattice over $\{a, b, c, d\}$ with *MChain*(⟨W, e, e, L⟩) in bold



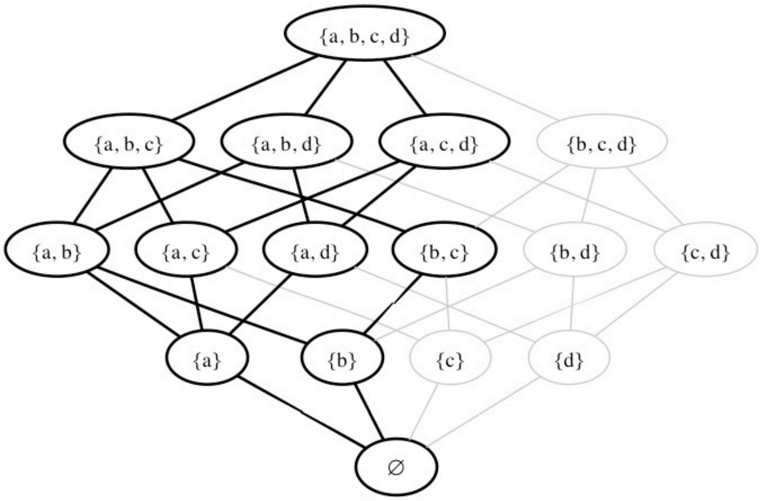
The Erc ⟨W, W, L, L⟩ produces a different sublattice of the power lattice, shown in Figure 5, one in which the ranking requirement of *a* or *b* dominates *c* and *d* is satisfied by all twelve maximal chains in the lattice. Again, linear extensions are read off of the lattice by proceeding *up* the lattice, starting at the empty set and proceeding to the top of the lattice.

Figure 5. The power lattice over $\{a, b, c, d\}$ with *MChain*(⟨W, W, L, L⟩) in bold



The previous two examples showed the construction and representation of a single Erc and its licit linear extensions in a power lattice. Of course, we are likely to be concerned with, not single Ercs, but *sets* of Ercs and the linear extensions consistent with those Erc sets. Construction of the lattice representation for a set of Ercs E proceeds similarly to the method outlined above: select all those maximal chains that are associated with linear extensions consistent with the Erc set. The collection of these maximal chains constitutes *MChain*(E). Below, in Figure 6, this has been done with the Erc set E = {⟨W, W, L, L⟩, ⟨W, e, e, L⟩}. Notice that this lattice is equal to the lattice produced from intersecting the previous two lattices, as maximal chains in the lattice produced from the Erc set E must both be present in the lattice produced from the Erc ⟨W, W, L, L⟩ and the lattice produced from the Erc ⟨W, e, e, L⟩}.

Figure 6. The power lattice over {*a*, *b*, *c*, *d*} with *MChain*({⟨W, W, L, L⟩, ⟨W, e, e, L⟩}) in bold



The manner in which *MChain*({⟨W, W, L, L⟩, ⟨W, e, e, L⟩}) was produced (that is, intersecting the sublattices of *MChain*(⟨W, W, L, L⟩) and *MChain*(⟨W, e, e, L⟩)) can be easily extended to any set of Ercs: simply intersect the corresponding sublattices for each of the *MChain*s for each Erc in the set, producing that lattice that contains all the maximal chains that represent the linear extensions consistent with the Erc set.

We have now defined a function from consistent Erc sets to subsets of the power lattice on the constraint set. This definition is given below.

Definition 1. ***MChain***
Given a consistent set of Ercs E over a constraint set C, define *MChain*(E) to be the union of all maximal chains in 2^C (the power set lattice on C) of linear extensions consistent with E.

As we have seen *MChain*(E) is the intersection of all *MChain*(α) where α runs over the Ercs in E.

Returning to a single Erc, call it α, we know how to construct *MChain*(α) as a sublattice of the power lattice. It turns out that our construction of *MChain*(α) yields a tight relationship between the nodes on *MChain*(α) and L(α), the constraints marked with L in α. No node in this lattice contains an element of L(α) if that node does not also contain an element of W(α), the set of constraints marked with a W in α. Informally, this is because each total order in *MChain*(α) satisfies the Erc α and therefore has each L-having constraint preceeded by at least one W-having constraint. This nodal relationship between W(α) and L(α) will be useful at a number of points throughout our discussion and so we prove this in the lemma below.

Lemma 1
For an Erc α and *MChain*(α), no node in *MChain*(α) contains an element of L(α) if that node does not also contain an element of W(α).

Proof: Suppose otherwise, so that there is a node, *n*, in *MChain*(α) that contains an element of L(α) but no element of W(α). This means that there is a ranking encoded in *MChain*(α) containing node *n* that ranks an L before any W. Now α requires that each element of L(α) be preceded by an element of W(α), a fact that is respected in the construction of *MChain*(α). Therefore we have reached a contradiction. QED.

In the next section we define what an antimatroid is and show that our *MChain* function can be viewed as a function from Erc sets to antimatroids.

3 Antimatroids

Antimatroids are formal objects that have been discovered and rediscovered in many different fields under many different names. For our current purposes the most relevant property of antimatroids is that they generalize the notion of a partial order in a way that corresponds precisely to what can be encoded by a set of Ercs.

Ercs that have only one W and one L encode dominance among pairs of constraints – we will call these *simple* Ercs.[2] Rankings described by sets of simple Ercs are exactly those that can be described as a partial ordering of the constraints (and concomitantly which can be represented with Hasse diagrams).

Ercs that have more than one W correspond to disjunctive conditions on rankings. For example, in ⟨W, W, L, e⟩: either *a* or *b* dominates *c*. This state of affairs cannot be described with a partial order. Though more expressive than partial orders, the sets of rankings described by Erc-sets are fairly tightly constrained; Ercs describe all the sets described by partial orders plus those describable with the addition of disjunctive statements like the one above, but they do *not* describe arbitrary collections of rankings.

Antimatroids were first described by Dilworth (1940) in the context of work on lattices. There are many different but equivalent formal definitions of antimatroids (for extensive discussions, see Korte et al., 1991 and Dietrich 1987). The definition that is simplest for our current purposes is one formulated in terms of *accessible set systems that are closed under union*. In the following sections each of these terms will be made explicit. We will start with *set systems* and *accessible set systems* before adding in the condition that they be closed under union.

3.1 Accessible Set Systems

Antimatroids are accessible sets systems that are closed under union. In this section we will define and explain each of the components of this definition and show how they relate to our *MChain* function from the previous section. We start with the definition of a set system, further refining it by defining an accessible set system. We will then see that our previously defined function *MChain*, into the power set lattice, maps into an accessible set system.

Definition 2. Set System

---

[2] An Erc with only one W and multiple Ls corresponds to a set of simple Ercs. E.g. ⟨W, e, L, L⟩ = {⟨W, e, L, e⟩, ⟨W, e, e, L⟩}.

A set system, denoted (G, $\mathcal{m}$), is a finite set G, a 'ground' set, along with $\mathcal{m}$ a collection of subsets of G (i.e., $\mathcal{m} \subseteq 2^G$).

Transparently, *any* subset of a power set is a set system. And, of course, the image of any Erc set under the map *MChain* is also a set system. We are proceeding, piece by piece, to transform *MChain* into a map into antimatroids.

We will first define an accessible set system, which is a set system with two additional properties. Those two properties are properties of *augmentation* and *removal*. Informally, each set in the system can be augmented by a single element producing another set in the system (except the ground set), and each set can have an element removed producing another set in the system (except the empty set). Below is a precise definition.

Definition 3. Accessible Set System
An accessible set system (G, F) is a set system having the additional properties of *augmentation* and *removal*, listed below.

> (*Augmentation*) If S ≠ G, there exists an x ∈ G, x ∉ S, and a set T ∈ F, such that T = S ∪ {x}.
> (*Removal*) For each set S ∈ F, if S is non-empty there exists an x ∈ G and a set R ∈ F that does not contain x, such that S = R ∪ {x}.

What this means is that given any set S in the system (other than F) it is possible to add some element in G to S and obtain another set in the system. And furthermore, if S is not empty it is possible to remove some element from S and thereby obtain another set in the system. The term 'accessible' refers to the fact that both the empty set and the ground set are *accessible* from any given set in the system via a sequence of single element additions or removals.

The sets S ∈ F for an accessible set system are called the *feasible* sets of the system.

If we consider the set of constraints in an optimality theoretic system to be a ground set G, then the power set lattice over G is an accessible set system (as the ground set here can be viewed as an arbitrary set, it should be clear that *any* power set lattice is an accessible set system). In fact, the sublattices we were considering in Section 2 corresponding to Erc sets are also accessible set systems. We can see this in the example below.

Consider again the Erc set E = {⟨W, W, L, L⟩, ⟨W, e, e, L⟩}. From above we know that *MChain*(E) is a subset of the power set on the four constraints {*a, b, c, d*} that encodes all linear extensions consistent with E. The four constraints {*a, b, c, d*} are our ground set G, and *MChain*(E) is our collection F of subsets of G. Trivially we can see that with our definition of G and F we have defined a set system[3]. To see that it is an *accessible* set system consider below

---

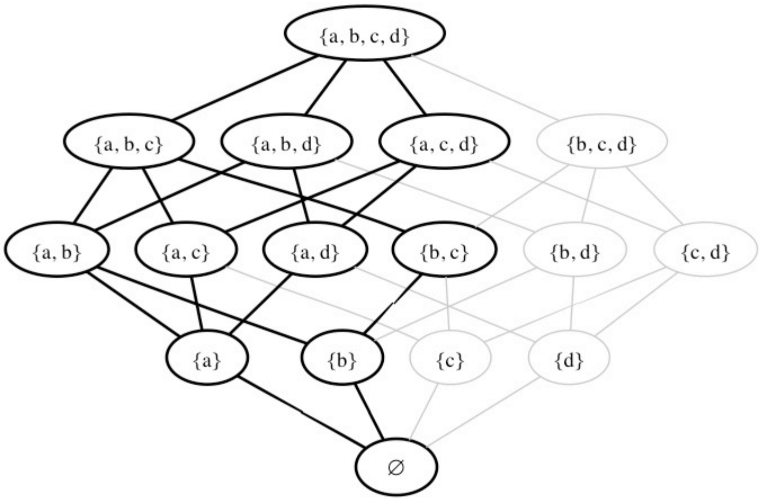
[3] It should also be obvious that if given *any* set of ERCs E then by defining the constraint set to be the ground set G, and *MChain*(E) to be F defines a set system – this immediacy comes from the triviality of the definition of a set system, *any* subset of the power set, not to be equated with the more structured *accessible* set system.

MChain(E) in Figure 7, as embedding in the power lattice over {a, b, c, d} (this is Figure 6 repeated here).

Accessibility means two things. First, given a non-empty feasible set S (i.e. S ∈ F) there is an element in S that can be removed from S and the resulting set will be another feasible set. This first property clearly holds for *MChain*(E). To take just one example, consider S = {a, b, d} ∈ F. By removing the element d from S one produces the set R = {a, b}, which is also an element of F. (Note that the element to be removed is dependent on S – removing b from S would have produced {a, d}, a set that is not in F.) We can quickly visually inspect *MChain*(E) to see that each non-empty set S in F satisfies this property.

Second, accessibility means that given a set S in F that does not equal the ground set one can *add* some element to it to produce another set in F. To be concrete, consider {a} ∈ F. One can add either b or c to {a} and get an element in F. As can be seen, all sets in F (except the ground set) satisfy this property.

Figure 7. The power lattice over {a, b, c, d} with *MChain*(E) in bold



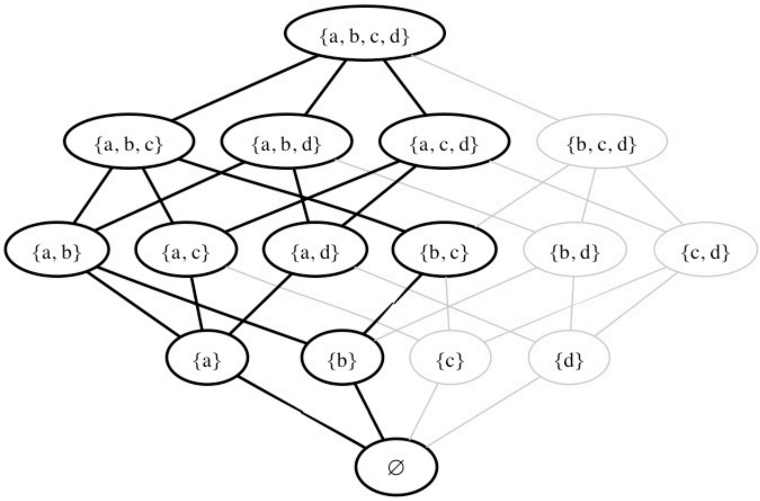
The fact that the *MChain*(E) associated with Erc set E forms an accessible set system for *any* Erc set E is fairly straightforward to show and will prove to be one core component of demonstrating that antimatroids are equivalent to Erc sets. We prove this fact below.

Lemma 2
For a set of consistent Ercs E, *MChain*(E) is an accessible set system.

Proof: Let G be the set of constraints from which the Ercs in E are constructed. Then F = *MChain*(E) is a subset of the power set of G, and (G, F) forms a set system. To show that (G, F) is an accessible set system, first let S ∈ F with S non-empty. F is composed of those maximal chains that are consistent with all of the Ercs in E, and F is the intersection of all *MChain*(E_i) where E = $\cup_i E_i$, where the E_is run over the Ercs in E. Returning to S, since S is in F, it is in some maximal chain that is in each *MChain*(E_i), call this maximal chain *max*. Since *max* corresponds to a linear extension, S represents the ranking of the first *k* constraints where /S|=*k*. Now, there exists R ∈ *max* where |R| = *k*–1 where, similar to S, R represents the ranking of the first *k* – 1 constraints. Clearly R and S differ in precisely one element as they are both elements of the single maximal chain *max*. Call this element x. So, we have shown that there exists some element x ∈ S so that S\{x} = R ∈ *max*. Now secondly, we need to show that if S ≠ G then there exists x ∈ G such that S ∪ {x} ∈ F. This follows in a nearly identical manner to what we just showed. S is in some maximal chain *max* that is a subset of F. Let /S|=*k* and since S ≠ G there exists T ∈ *max* where |T| = *k*+1. Clearly S and T differ by one element, call it x, and with that we have shown there exists x ∉ S such that T = S ∪ {x}. QED.

While the formal proof is necessary, an intuitive understanding grounded in an example exemplifying why it is true may be useful. The proof relies on the fact that any set S ∈ F is part of a maximal chain that corresponds to a linear extension that is consistent with each of the Ercs in the ERC set that F is built from. Consider our recurring example of an accessible set system (G, F), where G = {*a*, *b*, *c*, *d*} and F = *MChain*(E) where E = {⟨W, W, L, L⟩, ⟨W, e, e, L⟩}. Now consider the set S = {*a, c*} again. The ground set G and the empty set {} are accessible from S (in the sense one can traverse the lattice either up or down via single element addition or removal) because S is part of the maximal chain {{}, {a}, {a,c}, {a,b,c}, {a,b,c,d}}. Maximal chains, of course, correspond to linear extensions, in this case the linear extension *acbd*. This linear extension is consistent with both of the Ercs in E, as is made clear by quick inspection. As F is constructed from the union of maximal chains corresponding to consistent linear extensions, each element in F will have the desired accessibility property.

Figure 8. The power lattice over {*a*, *b*, *c*, *d*} with *MChain*(E) in bold



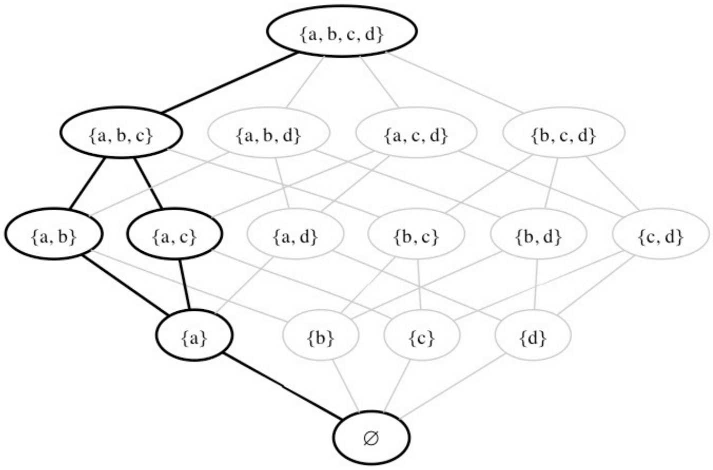
At this point we have demonstrated that any consistent Erc set can be represented as an accessible set system, and we are nearly at the point where we can define antimatroids and demonstrate that our representation in power sets of Erc sets are antimatroids. But we first need to note one final property about the accessible set system representation we have constructed.

Definition 4. Closed Under Union
A collection of sets, C, is said to be *closed under union* if given any two sets S, T ∈ C, then the union of those two sets is also in C.

So, for a set C, C is closed under union if for all S, T ∈ C, S ∪ T ∈ C. As we will show below, it turns out that the accessible set systems we constructed from ERC sets are closed under union.

Closure under union is easily seen in a small example. Consider the accessible set system constructed from the Erc set E = {⟨W, L, L, L⟩,⟨e, W, e, L⟩,⟨e, e, W, L⟩}. The set system is embedded in the power lattice below. Note that there are only two linear extensions consistent with these three Ercs: *abcd, acbd*, and these two can be easily read off of the accessible set system bolded in Figure 9.

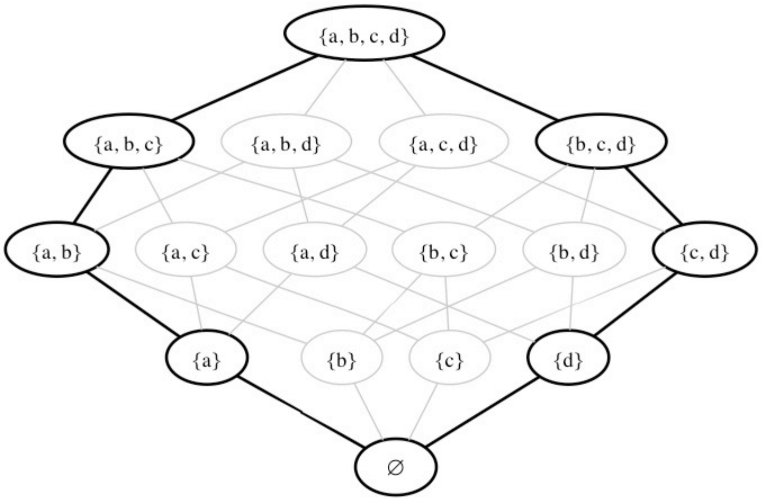
Figure 9. The power lattice over $\{a, b, c, d\}$ with *MChain*(E) in bold



Quick inspection shows that, indeed, *MChain*(E) is closed under union. Take any two elements of *MChain*(E), say {a, c} and {a, b}, and their union, in this case {a, b, c}, is an element of *MChain*(E). One can quickly do this for any pair of sets from the 6 sets composing *MChain*(E).

Though the maximal chains for the linear extensions of an ERC set are always closed under union, this property does not hold for arbitrary collections of linear extensions. Consider the two linear extensions *abcd*, *dcba* and their maximal chains, shown in figure 10 below. Selecting two sets in this sublattice and unioning them does not always yield another element in the sublattice, as can be seen by selecting {a} and {d}. The union of these two sets yields {a,d}, a set that is not in this sublattice. So why in this case is the set not closed under union? The answer to this lies with the Ercs that represent the linear extensions. In the previous example three ERCs (⟨W, L, L, L⟩,⟨e, W, e, L⟩,⟨e, e, W, L⟩) picked out the two linear extensions (*abcd, acbd*) that yielded the accessible set system. Here, there are *no sets of Ercs* that can pick out these two linear extensions and no others.

Figure 10. The power lattice over {*a*, *b*, *c*, *d*} with the maximal chains of *abcd*, *dcba* bolded



Closure under union in examples like the one above corresponds exactly to the distinction between sets of linear extensions that can be represented by Ercs and those that cannot. As we will show below, the accessible set systems built from sets of Ercs are closed under union.

Lemma 3
For a set of consistent Ercs E, the accessible set system *MChain*(E) is closed under union.

Proof: Let S, T ∈ *MChain*(E). We want to show that S ∪ T ∈ *MChain*(E). First note that, by Lemma 1, if either S or T contains a constraint in L(α) for some ERC α in E, that that set also contains a constraint in W(α). This property also holds for S ∪ T, since for any constraint in L(α) in S ∪ T there is also a constraint W(α) in S ∪ T, since S ∪ T contains all and only those constraints in S and T. So we have established that S ∪ T has the property that if S ∪ T contains a constraint in L(α) for some Erc α in E, then S ∪ T also contains a constraint in W(α). But any set with this property is an element of *MChain*(E) since having this property implies that a set is part of a maximal chain in which the ranking conditions for each of the Ercs in E is satisfied. QED.

We can now complete our definition of antimatroids and show that our function *MChain* maps sets of consistent Ercs onto an antimatroid.

Definition 3: Antimatroid
An antimatroid is an accessible set system that is closed under union.

Given this definition of an antimatroid[4], in terms we have spent some time becoming familiar with, we can define a new function from consistent Erc sets to antimatroids.

Definition 4. ***Antimat***

---

[4] There are numerous equivalent definitions of antimatroids. See Dietrich 1987 for a number of variants.

Given a consistent set of Ercs E over a constraint set C, define *Antimat*(E) = (C, *MChain*(E)) to be the set system with ground set C and feasible sets *MChain*(E).

It is a clear consequence of Lemma 3 that our newly defined *MChain*function does indeed map consistent Erc sets to antimatroids. We record this fact in Lemma 4.

Lemma 4. Given a consistent Erc set E, the function *Antimat*(S) = (C, *MChain*(E)), where C is the set of constraints from the Ercs in E, maps S onto an antimatroid.
Proof: This follows immediately from Lemma 3 and the definition of an antimatroid.

We have now defined a function from consistent Erc sets to antimatroids. We did so by showing that every consistent Erc set corresponds to an accesible set system closed under union, and now we will show that every accessible set system closed under union (i.e. every antimatroid) corresponds to a consistent Erc set. We will do so by defining a function from antimatroids to consistent Erc sets that is the inverse of our *MChain*function. To do this we first need to understand some of the formal properties of antimatroids better. It turns out that antimatroids can be characterized by what are called *rooted circuits*, and this characterization will allow us to construct the desired function from antimatroids to Erc sets.

3.2 Rooted circuits

A given antimatroid is uniquely defined by an associated set of antimatroids (see Dietrich 1987 and Korte et al. 1991 for discussion and proof). These defining antimatroids are called the *rooted circuits* of an antimatroid. As we will see below, there is a natural correspondence between a given rooted circuit and an Erc. By associating each rooted circuit of an antimatroid with an Erc we will be able to construct a function from antimatroids to Erc sets. We start though, with defining the rooted circuits of an antimatroid.

Defining the rooted circuits of an antimatroid is a two-step process, starting first with defining the *traces* of an antimatroid, and then selecting from an antimatroid's traces its rooted circuits. So, to start, given an antimatroid $A = (G, F)$, where G is the ground set and F are its feasible sets, and another set $S \subseteq G$, the trace of A on S, which we denote by $F{:}S$, is given by the following definition.

Definition 5. The trace of *A* on *S, F:S*.
For a given antimatroid $A = (G, F)$ and $S \subseteq G$, the trace of A on S, $F{:}S = \{f \cap S \mid f \in F\}$.

It is important to note that the trace of *A* on *S* immediately yields another antimatroid consisting of a ground set S and feasible sets *F:S*.

Turning towards a concrete example of a trace, consider the antimatroid *A* = (G, F) where the G = {a, b, c, d} and the feasible sets of A are below in figure 11 given by the bolded sets in the power set lattice over {a, b, c, d}. The antimatroid defined by these sets is the same as *Antimat*(⟨W, e, e, L⟩) (Figure 11 is a reproduction of Figure 4 above); being such it contains all of the maximal chains corresponding to linear orders consistent with the Erc = ⟨W, e, e, L⟩. Now, let *S* = {a,b}. We then construct the trace of *A* on *S*, denoted F:S. This is done by taking the set S and intersecting it with each of the feasible sets of A. Since the feasible sets of A contain, amongst others, {}, {a}, {b}, and {a,b} and we are intersecting each of these sets with S = {a,b}, the resulting trace consists precisely of these 4 sets, as is shown in Figure 12.

Figure 11. The power lattice over {*a*, *b*, *c*, *d*} with the antimatroid *A* = (G,F)[5] with its feasible sets in bold

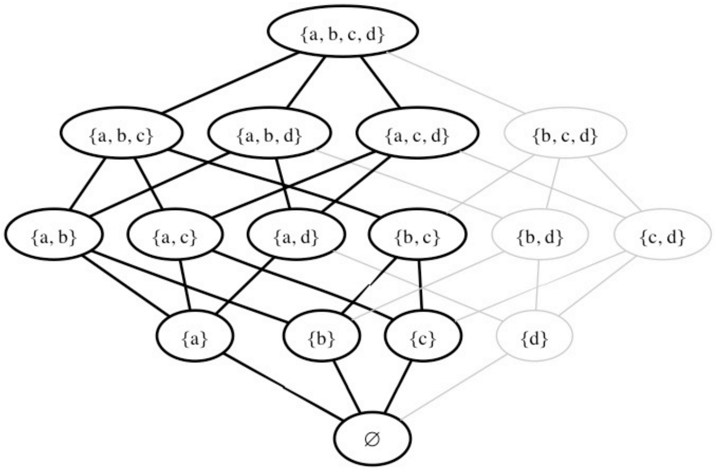
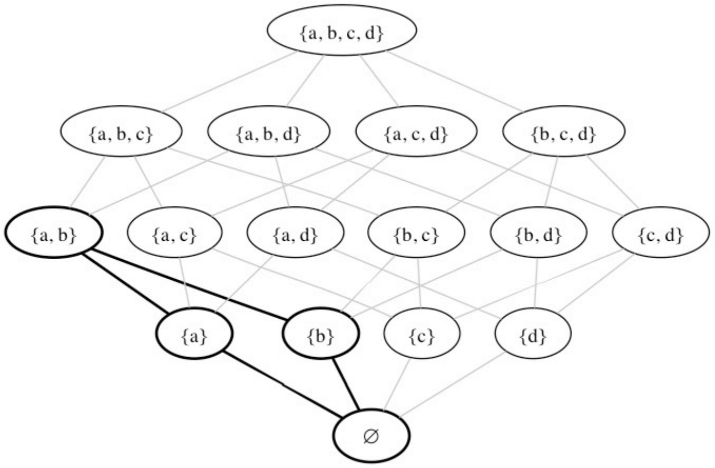


Figure 12. The trace of A on S, where S={a,b}.



A useful way of thinking about traces is that the trace of A on S precisely captures the ordering relations that the original antimatroid places on the constraints in S. So in this example where S={a,b} and our original antimatroid represents the Erc ⟨W, e, e, L⟩, as analysts we know that ⟨W, e, e, L⟩ places *no* restrictions on the relative ordering of a and b, that is, a can precede b or b can precede a. The trace F:S represents this fact by being an antimatroid on two constraints, {a,b}, one in which the maximal chains encode any ordering of a and b. Recall that maximal chains represent ordering by proceeding up the subset lattice from the empty set. Here the ordering a≫b is encoded by the sets {}, {a}, {a,b}, and the ordering b≫a by the sets {}, {b},

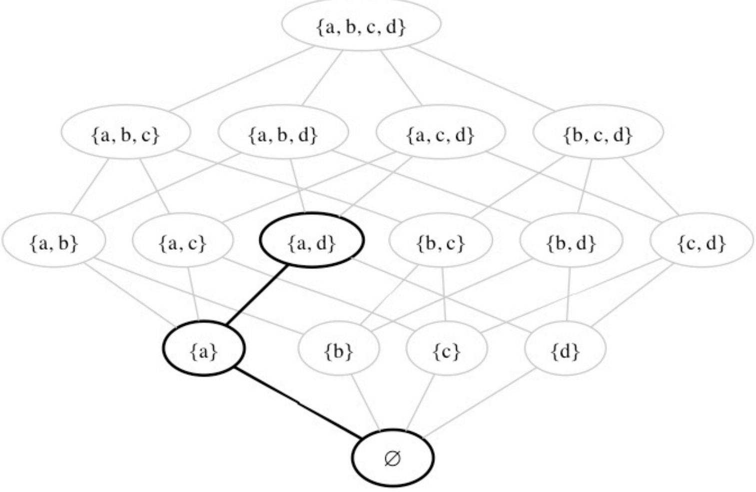
---

[5] This antimatroid is equivalent to *MChain*(⟨W, e, e, L⟩) depicted in Figure 4 above.

{a,b}. Both orderings are encoded in A:S, representing the fact that the original antimatroid A, interpreted as implementing the Erc ⟨W, e, e, L⟩, places no relative restrictions on the ordering of a and b.

Looking to another example, one in which there are restrictions on relative orderings, consider T = {a,d} and the trace of A on T, F:T, given below in Figure 13.

Figure 13. The trace of A on T, where T={a,d}.



In this example F:T represents one total order on a and d, namely a $\gg$ d. This codifies the fact that the original antimatroid, again interpreted as representing the Erc ⟨W, e, e, L⟩, requires that a precede d in any ordering that satisfies ⟨W, e, e, L⟩.

Some antimatroids place no restrictions on the orderings that they encode, as we have seen in F:S, shown above in Figure 13. Antimatroids of this type are said to be *free*. Equivalently an antimatroid A=(G,F) is *free* if the feasible sets of A consist of all possible subsets of G, that is, F = 2^G. This definition of a free antimatroid will play a crucial role in our rooted circuit definition.

Definition 6. A free antimatroid.
For a given antimatroid $A = (G, F)$, A is said to be *free* iff $F = 2^G$.

Having defined the trace of a subset of the ground, we are now in a position to define the *rooted circuits* of an antimatroid which, as stated above, uniquely determine it.

Definition 7. A rooted circuit of an antimatroid *A*.
For a given antimatroid $A = (G, F)$ and $S \subseteq G$, the trace of A on S, *F:S*, is a *rooted circuit* if for every T ⊂ S, T a proper subset of S, *F:T* is free and *F:S* itself is not free.

So, first, a rooted circuit of an antimatroid is a trace of some subset of G. Being a trace it itself is an antimatroid. What differentiates a rooted circuit from a run-of-the-mill trace is that every proper subset S defines a free trace, and the whole rooted circuit itself is not free.

As we have seen, the trace defined by S = {a,b}, shown in Figure 13 above, while it has the property that every proper subset of S defines a free trace, since F:S is free it is *not* a rooted circuit. Now, the trace defined by T = {a,d} *is* one. First, it is not free. Second, every proper subset of T defines a free trace – there are only three proper subsets of {a,d}, those being {}, {a}, {d}, and it is easy to check that they define free traces on A. So, because all proper subsets are free and *F:T* is itself not free, *F:T* is a rooted circuit.

Note that if the feasible sets of *F:T*, our rooted circuit example, were augmented by a single set consisting of {d}, then *F:T* would be free (since the feasible sets now consist of $2^{\{a,d\}}$). It turns out that every rooted circuit, for any antimatroid, can have its feasible sets augmented by a single set consisting of one element from the ground set, turning the rooted circuit into a free antimatroid. This single element is called the *root* of the rooted circuit. So a rooted circuit *F:S* consists of the feasible sets $2^S \setminus \{r\}$ (where $r \in S$, and is the root of *F:S*). Traces that are rooted circuits will be denoted *F:S*(r) to identify their roots. This fact of single augmentation to power set is proven in Dietrich 1987 and will be useful for us later, and so we record this with a labeled lemma.

Lemma 5. A rooted circuit F:G_0(r) has feasible sets $F = 2^{Go} / \{r\}$.
Proof: See Dietrich 1987.

As mentioned above, each antimatroid has a unique set of rooted circuits. This uniqueness allows us to define a function from antimatroids to sets of Ercs (our goal here) by defining a function from rooted circuits to Ercs. The antimatroid to Erc set function then arises naturally by mapping an antimatroid to the set composed of the Ercs that are the images of its rooted circuits. Of course, to do so, we need to associate with each rooted circuit an Erc.

3.3 Defining a function from Antimatroids to Erc Sets

Recall that for an antimatroid *A*=(*G*,F) and S⊆G, the rooted circuit *F:S(r)*, represents the ordering relations that obtain amongst the constraints in S, as represented by A. Further recall that if we remove any single element from S we obtain a free trace – this means that any proper subset of S, viewed as rankable constraints, can be ordered in any particular order. So what are the ranking restrictions encoded by *F:S(r)*? Well, we know that {r} is not a feasible set of *F:S(r)*, and hence the antimatroid *F:S(r)* (and the antimatroid A) does not permit the constraint r to be ranked before any of the constraints in S \ {r}. But we also know that if we rank any of the constraints in S \ {r}, the resulting set is free so that the remaining constraints are free to be ranked in any old older – that is, one of the constraints from S \ {r} must be ranked before r. This situation is easily represented by an Erc, one that has W's for each constraint in S\{r} and an L for r. We are now able to define our map from rooted circuits to Ercs.

For an antimatroid *A=(G,F)*, *S*⊆G, and rooted circuit *F:S(r)*, define a function, *RCerc*, from rooted circuits to Ercs on the constraint set *G* by *RCerc*(*F:S(r)*) = α, where α is the Erc defined as follows.

Definition 8. Given an antimatroid $A=(G,F)$, $S\subseteq G$, define the function *RCerc* from a rooted circuit $F{:}S(r)$ to Ercs over the constraints G by:

$W(\alpha)$[6] $= S\backslash\{r\}$

$L(\alpha) = \{r\}$

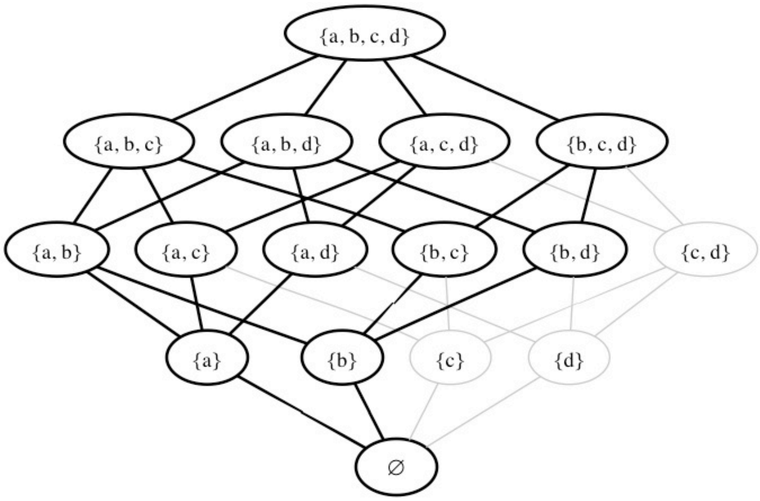
$e(\alpha) = G\backslash S$

We can immediately say (what we must) about this function, that it is indeed a function. The set of $W(\alpha)$, $L(\alpha)$, and $e(\alpha)$ partition the constraint set G, so that we are mapping into our desired co-domain, and furthermore, for a rooted circuit, $F{:}S(r)$, r is unique and so the function is well-defined.

Since a given antimatroid has a unique set of rooted circuits we can immediately extend this function to antimatroids, where for a given antimatroid A, and its unique set of rooted circuits RC, *RCerc*(A) maps to the set of *RCerc*(R), where R runs over the rooted circuits in RC.

We can see this map in action looking at the antimatroid given below in Figure 14. This antimatroid is the same as the antimatroid produced from *Antimat*(⟨W, W, L, L⟩,), and consequently this figure is the same as Figure 5 above.

Figure 14. The power lattice over $\{a, b, c, d\}$ with the antimatroid A = (G,F)[7] with its feasible sets in bold



This antimatroid has two rooted circuits $F{:}G_1(c)$ and $F{:}G_2(d)$, built from the sets $G_1 = \{a, b, c\}$ and $G_2 = \{a, b, d\}$ – all other strict subsets of {a, b, c, d} produce free traces (for example the members of {a, b} can occur in either order and hence the trace of {a, b} is free).[8] Under the *RCerc* map, these two rooted circuits are mapped to ⟨W, W, L, e⟩ and ⟨W, W, e, L⟩

---

[6] Recall that the set of constraints marked with a W in Erc α is denoted $W(\alpha)$, the set of constraints marked L is $L(\alpha)$, and those marked e, $e(\alpha)$.

[7] This antimatroid is equivalent to *MChain*(⟨W, W, L, L⟩).

[8] Freedom and rootedness do not exhaust the outcomes of trace production – a trace can be neither free nor a rooted circuit – see {a, b, d} with *Antimat*(⟨W, e, e, L⟩).

respectively. It immediately follows that *RCerc*(A) = {⟨W, W, L, e⟩, ⟨W, W, e, L⟩}, as our function on antimatroids maps an antimatroid to the set of Ercs its rooted circuits map to. Via quick inspection we can see that this Erc set is equivalent to the single Erc ⟨W, W, L, L⟩. As we have seen above in the discussion of *Antimat*(⟨W, W, L, L⟩) our original antimatroid *A* is the union of all maximal chains that satisfy the Erc ⟨W, W, L, L⟩, giving us hope that our two functions, *RCerc* and *Antimat*, are inverses of one another. In the next section we will prove that they are indeed inverses.

4 Antimatroid and Erc Set Equivalence

In this section we will prove that the two functions *RCerc* and *MChain* are inverses of one another and that they are also homomorphisms, preserving the entailment relations between Erc sets and the containment relations between antimatroids.

4.1 Proof overview

We will start our proof by showing that given an arbitrary antimatroid *A*=(*G,F*) the map *MChain* is the inverse of *RCerc*, so that *Antimat*(*RCerc*(*A*)) = *A*. Showing this is equivalent to showing that if *A* has rooted circuits *RC*(*A*) and the antimatroid *B*=*MChain*(*R*), where *R*={*RCerc*(F:G_0(r)) | F:G_0(r) ∈ *RC*(A)}, then *A*=*B*. This is equivalent because *B* is the image under *RCerc* of the Erc set *Antimat*(*A*). Once we have established that *MChain*is the inverse of *RCerc*, establishing that *RCerc* is the inverse of *MChain*will follow easily, as will that they are homomorphisms.

Our proof of *RCerc*'s inversibility requires a number of intermediate results, possibly obscuring the overall logic of the argument, and so a list of what will be proven is given below.

Proof overview:

Step 1: Every antimatroid is composed of the maximal chains of a set of total orders.
Step 2: Given a total order λ such that *MChain(*λ) ∈ A, then *MChain*(λ) ∈ B.
Step 3: Given a total order λ such that *MChain(*λ) ∈ B, then *MChain*(λ) ∈ A.
Step 4: Conclusion. *A* and *B* are built from the same total orders, so they are the same antimatroids.

In our first step, proven in Lemma 5 below, we show that every antimatroid is the union of the maximal chains of total orders on the ground set (previously, in the construction of *RCerc* we only showed that the union of maximal chains is an antimatroid, leaving aside whether *every* antimatroid can be constructed in such a manner).

Lemma 6. Let A=(G,F) be an antimatroid. Then there is a set of total orders L of G such that F={*MChains*(λ) | λ ∈ L}.

Proof: Let A=(G,F) be an antimatroid and let f ∈ F, with f ≠ ∅. Then because A is an antimatroid there is an a_1 ∈ f such that f \ a_1 ∈ F. If f \ a_1 ≠ ∅ this process of single element removal can be repeated producing f \ a_1 \ a_2 ∈ F. This process can be repeated until the empty set is reached, sequentially producing elements a_1, a_2, … a_k. These elements, in reverse order, will be the initial sequence of our total order. Now, if f ≠ G, because A is an antimatroid there is a b_1 ∈ G, b_1 ∉ f, such that f ∪ {b_1} ∈ F. Paralleling element removals, this process can be repeated adding a single element to a given feasible set until one reaches G. Let us call these elements b_1, b_2, … b_j. We have just produced a total order $\lambda = a_k a_{k-1} \ldots a_1 b_1 b_2 \ldots b_j$ such that *MChain*(λ) ⊆ F. Furthermore, we have produced a total order that includes our arbitrarily selected feasible set f. So, if we union all of the maximal chains produced for all of our feasible sets we have produced L. QED.

Having established Step 1, we move onto Step 2 and show that if there is a total order λ such that *MChain(*λ) ∈ A, then *MChain*(λ) ∈ B. We do this in Lemma 6 below, showing that a λ whose maximal chain is in A also satisfies all the Ercs from the rooted circuits of A. And since satisfying these Ercs is a guarantee of having a maximal chain in B, we will have established Step 2.

Lemma 7. Given an antimatroid A=(G,F) and λ a total order such that *MChain*(λ) ⊆ F, then *MChain*(λ) satisfies all Ercs *RCerc*(F:G_0(r)) where F:G_0(r) runs over the rooted circuits of A.
Proof. Let A=(G,F) be an antimatroid and λ a total order such that *MChain*(λ) ⊆ F. Looking to produce a contradiction, assume that there is a rooted circuit of A, say F:G_0(r), such that λ does not satisfy *RCerc*(F:G_0(r)). Now λ defines a total order on the n constraints under consideration: label this order $a_1 a_2 \ldots a_k r a_{k+2} a_{k+3} \ldots a_n$. Since λ does not satisfy *RCerc*(F:G_0(r)), W(*RCerc*(F:G_0(r)))∩{a_1, a_2, …, a_k} = ∅ (intuitively, λ not satisfying the Erc *RCerc*(F:G_0(r)) means that the constraint r appears in λ before each of the W's of *RCerc*(F:G_0(r), hence the set of constraints ranked before r in λ, {a_1, a_2, …, a_k}, contain no W's of *RCerc*(F:G_0(r)). But this will lead inexorably to a contradiction. Indeed, since *MChain*(λ) ⊆ F, the set S={a_1, a_2, …, a_k, r} ∈ F, that is, the first k+1 constraints ranked in λ comprise a feasible set of F. Since S contains no W's of *RCerc*(F:G_0(r)), S ∩ G_0 = {r}. But this is our sought after contradiction since {r} = S ∩ G_0 is an element of the feasible sets of F:G_0(r), and by Lemma 5, the feasible sets of F:G_0(r) do not contain the set {r}. QED.

Our final difficult step, Step 3, (though not our final step), requires a lemma already established in the antimatroid literature. An antimatroid's feasible sets are characterizable by their interaction with the rooted circuits feasible sets. This is made more precise in the lemma below.

Lemma 8: Let A=(G,F) be an antimatroid with rooted circuit collection RC(A) and let S ⊆ G. Then S ∈ F iff no rooted circuit in RC(A) meets S only on its root. Equivalently: F={S⊆G | F:G_0(r)∈RC(A) ⇒ S∩G_0 ≠ {r}}.
Proof: See Dietrich 1987.

We now can take on Step 3, showing that given a total order λ such that *MChain(*λ) ∈ B, then *MChain*(λ) ∈ A.

Lemma 9. Let A=(G,F) be an antimatroid. If a total order λ is such that λ satisfies *RCerc*(F:G_0(r)) for every rooted circuit of A, then *MChain*(λ) ⊆ F.
Proof: Let S ∈ *MChain*(λ). If we can show that S ∈ F we will have shown that *MChain*(λ) ⊆ F. Consider a rooted circuit of A, call it F:G_0(r). First, suppose that S contains r. By Lemma 1 S

must contain some other element of G_0 (recall that Lemma 1 states that for an Erc α, if $S \in$ *MChain*(α) and S contains some element of L(α), then S contains some element of W(α)). Therefore $S \cap G_0 \neq \{r\}$. Now suppose that S does not contain r. Then clearly $S \cap G_0 \neq \{r\}$. Now since the rooted circuit F:G_0(r) was selected arbitrarily, we can conclude that $S \cap G_x \neq \{r\}$ for any set $G_x \subseteq G$ where F:G_x is a rooted circuit of A. But this means, because of Lemma 8, that $S \in F$. QED.

Step 4 follows immediately. The antimatroid B which is the image of *RCerc*(A) under *MChain*is the same antimatroid as A since they are built from the same total orders. This allows us to conclude that *MChain*is the inverse of *RCerc*.

Theorem 1. *MChain*is the inverse of *RCerc*, and so *Antimat*(*RCerc*(A)) = A for any antimatroid.
Proof: This follows immediately from Steps 1 – 4.

It turns out that these lemmas are enough to also prove that *RCerc* is the inverse of *Antimat*.

Theorem 2. *RCerc* is the inverse of *Antimat*, and so *RCerc*(*Antimat*(E)) = E for consistent Erc sets.
Proof: Given a consistent set of Ercs E, *Antimat*(E) is that antimatroid that contains all maximal chains satisfying E. Now *RCerc*(A) maps an antimatroid to a set of Ercs whose satisfying total orders are those whose maximal chains constitute A by lemma 7 and lemma 9. But this means that *RCerc*(*Antimat*(E)) maps to a set of Ercs that are logically equivalent to E. QED.

Being inverses of each we have established a bijection between consistent Erc sets and antimatroids.

4.2 Antimatroids and consistent Erc sets are isomorphic

Bijections between sets abound. What makes the bijection defined by RCerc and MChaina useful bijection is that it maintains the relationship that obtains between Erc sets and it maintains the relationship that obtains between antimatroids. In the mathematical nomenclature this bijection is an isomorphism.

The natural structure between Erc sets is one of entailment. An Erc set E entails an Erc set E' if all linear extensions that satisfy E also satisfy E' (See Prince 2002 for extensive discussion). A similar relationship between antimatroids obtains. An antimatroid A entails an antimatroid A' if all maximal chains in A are also in A'. Given these internal structures on Erc sets and antimatroids, we can easily determine that both *RCerc* and *MChain*maintain these entailment relations.

Theorem 3. Given consistent Erc sets E and F, if $E \vDash F$, then *Antimat*(E) $\vDash$ *Antimat*(F).
Proof: This follows immediately from the construction of *MChain* and the definition of antimatroid entailment.

Theorem 4. Given antimatroids A and B, if $A \vDash B$, then *RCerc*(A) $\vDash$ *RCerc*(B).

Proof: This follows immediately from the construction of *RCerc* and the definition of Erc set entailment.

With Theorems 1 – 4 we have established that antimatroids and sets of consistent Erc sets over the same ground set/constraint set are isomorphic. This allows us to reason about one using the other, and to port results freely between the two sets of objects.

5 Conclusion

We have established that each consistent Erc set has an equivalent antimatroid over the same ground set/constraint set and vice versa. The maps *RCerc* and *MChain*are inverses of one another and they maintain the inherent structure in their mappings, demonstrating that given a consistent Erc set, one can reason about it using its corresponding antimatroid and vice versa. We hope that this equivalence will allow linguists to import results from the large body of work done on antimatroids and lead to fruitful cross-discipline collaboration.

References

Dietrich, Brenda, 1987. "A Circuit Set Characterization of Antimatroids", *Journal of Combinatorial Theory*, Series B 43, pp 314–321.

Dilworth, Robert, 1940. "Lattices with unique irreducible decompositions", *Annals of Mathematics* 41, pp 771–777.

Korte, Bernhard; Lovász, László; Schrader, Rainer, 1991. *Greedoids*. Springer-Verlag, pp 19–43.

Monjardet, Bernard, 1985. "A use for frequently rediscovering a concept", *Order* 1: pp 415–417.

Prince, Alan, 2002. Entailed Ranking Arguments, ROA-500. http://roa.rutgers.edu/.

Prince, Alan, forthcoming. A paper on the equivalence of antimatroids and Erc sets.

Prince, Alan; Smolensky, Paul, 1993/2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Blackwell Publishers (2004). Technical Report, Rutgers University Center for Cognitive Science and Computer Science Department, University of Colorado at Boulder (1993).

Riggle, Jason, 2010. Sampling Rankings, ROA-1075. http://roa.rutgers.edu/.