

Reconstructing Combinatorics

Daniel Nikpayuk

Contents

Preface	iii
I Contextualizing Enumerative Media	1
1 Some Set Theory	3
1.1 Review	3
2 Some Graph Theory	5
3 Mathematical Grammar	7
3.1 deconstruction	7
3.2 reconstruction	8
3.3 The Notation	8
3.3.1 an example	9
II Narrating Enumerative Media	11
4 Choices	13
4.1 Grammar	13
4.2 Declarations	13
4.2.1 General Choice	13
4.2.2 Extending Choice	14
4.2.3 Composing Choice	15
4.2.4 Reconciling Choice	15
4.3 Enumerations	15
5 Rankings	17
5.1 Grammar	17
5.2 Declarations	17
5.2.1 General Ranking	17
III Deconstructing Enumerative Media	19
6 Inclusion-Exclusion	23
7 Pigeonholes	25

IV Reconstructing Enumerative Media	27
8 Generating Structors	31
8.1 Formal Power Series	31
8.2 Generating Functions	31
8.3 Reencoding Choice	32
8.4 The Binomial Theorem	32
9 Recurrence Relations	33

Preface

This book was written for many reasons. The first and main one being that doing so has brought the author some kind of contentment. Why take up something of such a tedious nature after all if you don't really enjoy it? The second and apparently more useful reason is to provide a standardization and axiomatization of the foundations of combinatorics—intending to hold the same level of rigor as observed by other branches of mathematics; this being something I do not currently see in other books on the topic.

Regarding the nature of “contemporary” combinatorial pedagogy, I have seldom found a book on combinatorics that was as intuitive to understand for myself as it likely was for its author. In no way or form do I say this as disrespect—there are many very eloquently and elegantly written books out there—rather I say it to expose my own deficiencies: I am envious of those combinatorialists who can break down combinatorial problems with *word associations* alone. I, on the other hand, have had to write an entire book so as to build an artificial intuitive framework just to solve some basic combinatorial problems, whilst the seamlessly talented mathmagicians could solve those and much harder naturally, and with much ease.

With that said, the aim of this book is not to provide any new plot twists to the standard introduction, but instead to analyse the underlying assumptions that currently allow us to communicate these concepts and their *theorems*. Another way of saying this, and it might sound like a detraction of one's own text,¹ is that the overall intention of this book is not as much to provide the reader with the ability to solve combinatorial problems, rather to provide the user with the ability to create and communicate a consistent language for the dialogue of their solutions.

It is my own belief—and to use literary terms—that combinatorics studies the narratives of *representation* and *arrangement*. As far as themes go, the story of combinatorics may be said to frequently focus its attention on matters of *existence*, *enumeration*, and *optimization*. Context stems from the standard axiomatic approach, though the plot is another matter (as one who delves deeper will come to understand); I am willing to say though that there are very few possible plots in which these main themes don't intersect. I will relegate such problematics to another discourse, and so with that said, it should be noted that given this is intended as an introduction to the subject, it largely aims at focusing more on matters of enumeration.

The reader may be wondering why in the above paragraph I chose to put mathematical matters in the terms of literary language. My claim for doing so is a personal one: It has been my observation that the one area of mathematics that even us math people have a hard time with is combinatorics; As mentioned above, I am no exception to this. It is my own observational correlation then that the few math people who are good at combinatorial thinking are also good at various styles of literary thinking. Without going into the long history that brought me to this conclusion I will say that it has been reinforced by my own studies of the humanities and thus I believe such “inter-disciplinary” borrowing will be fruitful in here and in the future.

I will conclude by saying that to do what I've otherwise set out to accomplish, I feel it necessary to break from the traditional approach to the teachings of combinatorics. The strategy is to introduce the following:

- syntax: the notation,
- semantics: the concept definitions, and most importantly
- grammar: the ability to navigate and translate combinatorial problems using bijective associations.

To elaborate: The first part of this is to formally define the notions of a combinatorial *sentence* as well as *formula*. A necessary part in doing this will be the construction of a notation not currently used in mainstream mathematics. Following this, I will begin to *reconstruct*² the underlying concepts used in the expression of current

¹It is not.

²I am borrowing from the literary notion of *deconstruction*. I could have titled this book “Deconstructing Combinatorics” but I felt the related word and concept of *reconstruction* held more inclusive and positive connotations.

combinatorial language. All of this will follow axiomatically and will be embedded with many examples to show use of the grammar. The main tool as it turns out is that of *bijective associations*.

Part I

Contextualizing Enumerative Media

Chapter 1

Some Set Theory

To maintain a certain level of formality, I will include some basic set theory as well as some basic graph theory. For those who know these two theories quite well—I say this to you: Please skim over these two sections with some attention, I have made minor modifications to notation and definitions (though nothing that can’t be equated to a logical equivalence towards any of the mainstream notations and definitions of the respective subjects).

1.1 Review

To begin, the *difference* of two sets A, B written $A \setminus B$ is defined to be a set containing exactly those elements of A that are not in B .

An *ordered pair* (a, b) is defined as the following set:

$$\{\{a\}, \{a, b\}\}$$

A *relation* R is a collection of ordered pairs. The set of “left-coordinates” of the pairs of R is called the *domain* of R and is written as $\text{Dom}(R)$. Symmetrically defined is the *range* of R , written $\text{Ran}(R)$, being the set of “right-coordinates” of the pairs of R . A relation R is said to *act on* (or “be over”) a set X if $\text{Dom}(R) \cup \text{Ran}(R) = X$.

An *equivalence relation* of a set is a relation acting on a set X that is: *reflexive*, *symmetric* and *transitive*. This is to say:

1. (reflexivity) For all $r \in X$, (r, r) is in R .
2. (symmetry) If $(r, s) \in R$, then $(s, r) \in R$.
3. (transitivity) If $(r, s) \in R$ and $(s, t) \in R$, then $(r, t) \in R$.

An ordering of a set is said to be *partial* if some elements are not comparable, and is otherwise said to be *total* or *linear*. A subset of a partially ordered set which is linearly ordered is called a *chain*. Within the realm of ordered sets, the difference between a *minimum* element and a *minimal* element is that a minimum element is comparable to every other element (totally ordered set) and is still the *least* element, whereas a minimal element is a least element but isn’t necessarily comparable with the all of the remaining elements (applicable to partially ordered sets).

A *mapping* (or *function*) from A to B , written $f : A \rightarrow B$ is a relation R_f such that $\text{Dom}(R_f) = A$; $\text{Ran}(R_f) = B$, and if $(x, y_1), (x, y_2) \in R_f$, then $y_1 = y_2$.

Let $\mathbf{x} : n \rightarrow X$ be a mapping from $\{0, 1, \dots, n-1\}$ to some range X . Then \mathbf{x} is called an n -tuple over X and is written as $\mathbf{x} = (x_1, x_2, \dots, x_n)$ where the x_k is to mean $\mathbf{x}(k)$. If the domain is not specified at a given time, \mathbf{x} will simply be called a *tuple* (over X).

let $\mathbf{x} = (x_1, x_2, \dots, x_n)$ be some n -tuple. As a notational convention, in order to represent x_1 from the view point of \mathbf{x} , I write $\mathbf{x}_{|1|}$ which indicates the *first coordinate* of \mathbf{x} . Moreover, I write $\mathbf{x}_{|2|}$ to represent x_2 , the *second coordinate* of \mathbf{x} , and in general $\mathbf{x}_{|k|}$ to represent x_k , the k^{th} coordinate of \mathbf{x} .¹

The *product* of an *indexed system of sets* $\langle S_i \mid i \in I \rangle$ is the collection of all functions $f : I \rightarrow \bigcup_{i \in I} S_i$ such that $f(i) \in S_i$. This is a way of defining an “abstraction” of a cartesian product of sets.

¹This notation is worth introducing here as it hints at a more important notation I will introduce later on and use throughout this book.

Definition (Well Ordered Set): A set S with an ordering imposed on it is said to be well ordered if:

1. the ordering is linear.
2. every nonempty subset $R \subseteq S$ has a least element.

Chapter 2

Some Graph Theory

As stated earlier, I will here modify the better known graph theoretic definitions slightly to suit my needs, the semantics are the same, it's simply the wording that is different. Again, for the reader better experienced in these matters, it is still worthwhile to at least quickly look over.

Definition (Graph): A graph G is a pair of sets (V, E) where V is called the vertex set (of vertices of G), and E is called the edge set of G , and is such that it consists of unordered pairs of vertices from V .

When two graphs G, H (or possibly more) are in a given context, to distinguish between their vertex and edge sets one writes V_G, E_G and V_H, E_H .

I'm not going to worry or bother with "multisets," or related concepts, which would entail an explanation about loops and multiple edges, the graphs I'm interested in are *simple*.

Given an edge $e = \{v_1, v_2\} \in E_G$ of some graph G , the vertices $v_1, v_2 \in V_G$ that compose it are said to be *incident* to e . Moreover, with respect to each other, the vertices v_1, v_2 are said to *adjacent*.

If G is some graph with $v \in V_G$, then the *degree* of the vertex v , written as $\deg(v)$ is equal to the number of edges $e \in E_G$ which are incident to it.

Definition (Subgraph): Let G be a graph, a graph H is said to be a subgraph of G if $V_H \subseteq V_G$ and $E_H \subseteq E_G$.

Reader beware, here is where I have altered the definitions just a little. I have gone about the definition of a "path" in a slightly different way, and although it might appear to be more complicated, it is intended to payoff later on.

Definition (Ordering of a Graph): Let G be a graph, the ordering $\text{Ord}(G)$ is the set such that for every edge $\{v_1, v_2\} \in E_G$ (with $v_1, v_2 \in V_G$), the *ordered* pairs (v_1, v_2) and (v_2, v_1) are exactly those elements in $\text{Ord}(G)$.

The members of $\text{Ord}(G)$ are called *directed edges*.

Definition (Path): Let G be a graph, let $v_1, v_2 \in V_G$. A sequence (p_1, p_2, \dots, p_n) with $p_1, p_2, \dots, p_n \in \text{Ord}(G)$ such that $(p_s)_{|2|} \neq (p_t)_{|2|}$ for any $s \neq t$ is called a path from v_1 to v_2 if the following conditions are satisfied:

1. $(p_1)_{|1|} = v_1$
2. $(p_i)_{|2|} = (p_{i+1})_{|1|}$ for $1 \leq i < n$
3. $(p_n)_{|2|} = v_2$

In particular, a path is called *closed* if $v_2 = v_1$ and is otherwise called *open*.

Definition (Connected Graph): A graph G is called connected if given any two vertices $v_1, v_2 \in V_G$ with $v_1 \neq v_2$, there exists a path from v_1 to v_2 . A graph G that is not connected is called disconnected.

The removal of a vertex v from a graph G , written as $G - v$, is defined as the removal of v from the vertex set V_G and any incident edges of v from E_G . Similarly, the removal of an edge e from E_G (but not the vertices incident to it) will be written as $G - e$.

Definition (Disconnect Vertex): Let G be a graph and v a vertex. If there exists some connected subgraph H with $v \in V_H$ such that $H - v$ is disconnected, then v is called a disconnect vertex of G .

Similarly, an edge e of G is called a *bridge* if its removal causes some connected subgraph H —to which it belongs—to become disconnected.

Here I introduce a two part “process” definition which is (as above) intended to be useful later on:

Definition (Open Partition): Let G be a connected graph possessing a disconnect vertex v . Let $\{H_i\}_{i \in I}$ be the collection of subgraphs obtained by the removal of v , then $\{H_i\}_{i \in I}$ is called an open partition of G .

Definition (Closed Partition): Let G be a connected graph possessing a disconnect vertex v . Let $\{H_i\}_{i \in I}$ be an open partition of G using v . If one were to unite v and its appropriate edge(s) back onto to each individual subgraph H_i written as H'_i , then the collection $\{H'_i\}_{i \in I}$ is called a closed partition of G .

It’s worth noting a closed partition is also a collection of subgraphs of the original. Both types of graph partitions aren’t partitions in the set-theoretic sense, but they’re pretty close; one being disjoint whose union is almost the full graph, and the other whose union is the whole, but isn’t exactly disjoint... but close.

Definition (Cyclic Graph): A graph G is said to be cyclic if it contains a closed path, in which case, such a path is also called a cycle. Furthermore, a graph G is said to be acyclic if it contains no cycles.

Definition (Tree): A tree T is a connected acyclic graph.

A vertex l of a tree T is called a *leaf* if $\deg(l) = 1$, all vertices of T that are not leaves are called *nodes*.

The reader will see later on that I am only interested in trees possessing at least one node.

Here ends the review.

Chapter 3

Mathematical Grammar

Currently in the literature, combinatorial arguments are constructed as interrogative statements: “Does *such-and-such* structure exist?” “How many ways are there to...?” etc. In my experience—and certainly I am loading this claim with a lot of individual bias—such free-form wordplay is often confusing and occasionally unstable. To successfully address such issues arising from these casual or “intuitive” combinatorial arguments, we will need a shift in perspective.

The problem generally arises by *procedurally* constructing the question before the knowledge of a solution. One of the more successful and well documented ways to avoid such pitfalls of procedural arguments is to *avoid them altogether*. Just kidding. Rather, if one relies on “existential” arguments assuming the existence of a structure based on some known criteria beforehand, rather than assuming the construction of a structure based on some other known criteria after the fact, the shift in perspective is often enough to establish one is making claims spanning more than just a void. Such safeguards are sometimes a bit more tedious but are worthwhile in that they tend to reduce contradictions and circular arguments.

In order to change the perspective from that of an interrogative or imperative statement to a declarative one, we must break down a few assumptions about what notation is and can be.

3.1 deconstruction

First consider the following:

$$'2 + 3'$$

This is all too common notation read as “two plus three.” This is what I would call *object-oriented* notation.¹ I will break down what I mean in parts. First, if one was prompted with this statement, they may evaluate it computationally; in the closest sense possible, it could in such a context be considered an imperative statement—a command (the evaluation resulting as ‘5’). This is not the perspective we are aiming for. Next, if again prompted, we may consider it an interrogative statement. Someone has questioned what this means, and if we are good literary analysts, we may wonder under what context this question has been posed, so as to interpret and devise a response. In the worst-case scenario, without further context we may reduce the question again to evaluating (the answer of course being ‘5’). Again, even in the best case-scenario with full context, this style of perspective is not the aim here.

The aim here then is to take a statement such as ‘2 + 3’ and to interpret it as nothing more than a declarative statement; one that simply records what has transgressed. Here is where the perspective begins to develop, but we need to digress just a little further... just a little further into the realm of linguistics:

To build my case and my claim, it will be helpful to analyse this declarative statement ‘2 + 3’ using some additional linguistic tools. In particular, we are interested in the syntactic relationships within, but to go any further we will need to look more closely at the statement itself.

To start us off, it would help to ask a question: Why have we left out the answer ‘5’?

Since we are more interested in the subtleties of syntax than the subtleties of semantics, and given this statement is to be interpreted as a *data structure*, the answer can be nothing more than: “the information is there, it’s just implied.” Thus, we could expand the statement out in words to be: “Five is computed, by addition, from two using three.” Admittedly, this is a big jump from “two plus three” but the translation is not unreasonable given what we

¹not to confuse it with “object-oriented” programming languages such as (of this date) C++ or Java.

are interested in, and this particular form makes clearer the syntactic relationships within. “Five” is the *subject*, “is computed, by addition” is the *verb* (with an adverbial modifier), “two” is the *direct object*, and (to keep things simple) “three” is the *indirect object*.

It is from this perspective I am willing to claim that ‘ $2 + 3$ ’ is object-oriented; if one were to read this statement this way, then this message—this *container of information*—can be said to favor the objects of this linguistic sentence.

This concludes the deconstruction portion of the notation. On to the reconstruction.

3.2 reconstruction

Now that we know what we’re working with, let’s look at this information in another way. It will be of more interest and use to consider notation that favors the verb—*verb-oriented* notation. In an object-oriented sentence, the objects are considered to exist first and so privilege is given to them; they are at the forefront of retrieval within the message. If the object is given first, then the relationships between the objects is communicated afterwards, the *structure* is communicated afterwards. With verb-oriented notation, it is the other way around, the structure is communicated first, and then the objects within.

As far as translation goes, such a notation is the destination, and the current object-oriented notation is the origin, and so we are only missing the steps in-between. As far as this goes, there are different possible approaches, but the one I have taken relies on graph theory. As best I can tell, Combinatorics and Graph Theory are largely inseparable, and so I would have introduced it at some point within this book. It is of course for the purposes of this notation I have introduced it so early on.

3.3 The Notation

Definition (Clause): A clause C is a pair of sets (G, M) such that G is a graph, and $M : S \rightarrow A$ is a mapping where its domain is a subset of the vertex set of G , which is to say $S \subseteq V_G$.

I extend the notions of *subgraphs*, *open* and *closed partitions* from graphs to clauses, the definitions themselves should be apparent.

Definition (Sentence): A clause $U = (T, M)$ is called a sentence if its graph T is a tree containing a node $n \in V_T$ such that:

$$\text{Dom}(M) = V_T - \{n\}$$

In such a case, n is called the *root* of the sentence, and is given the special character r_U .

If the context is clear the root r_U may be written without its subscript: r . It is worth noting that although the root does not belong to the domain, there exists a unique path from the root to every other vertex that does belong (which is infact every other vertex of the tree). This motivates the following notation:

With the verb as the foundation of the notation, we need to navigate our access to the elements of this structure. It is this approach that leads us additionally towards a “path-oriented” supplementary notation.

Given a sentence $U = (T, M)$ and a path p from its root r to some vertex b , I write

$$U_{/p[b]}$$

to mean

$$M(b)$$

though considering that b is implied within p already, I may—depending on the purpose—simply write it as:

$$U_{/p}$$

which is to say that $U_{/p}$ is the image of b under M . Such an image $U_{/p}$ of a sentence will be called an *argument* of U . As well, the vertex b that is “incident” to the path p will be called the *branch* of p . Finally, the set of such paths p_i from the root to a branch will be called the *syntax* of U and denoted as $\text{Syn}(U)$. When the context is clear, it will more often be written in the following way:

$$U : \Sigma_{i \in \text{Syn}(U)} p_i$$

and especially when the order of syntax is small I will most assuredly write it as:

$$U : p_0 + p_1 + \dots + p_n$$

As to why, this will be discussed further below.

I am not finished here though, if one inverts the order of the edges within the path p to obtain \bar{p} , one may write

$$[b]_{\bar{p}} \setminus U$$

to mean the same thing as

$$U_{/p[b]}$$

the difference here is a subtle change from *verb-orientation* to *object-orientation*, which is conceptually more useful for formula manipulations—more will be explained later. Finally, if the context is clear, I will “hide” the reference U and subscript the path p resulting in:

$${}_p[b] \quad \text{or} \quad [b]_{\bar{p}}$$

such notational definitions will be particularly important throughout the remainder of this article: don’t worry yet about how they’re applied, this is what much of the article is about.

A sentence S is called *simple* if the root r_S is its only node.

Definition (Base): Let B be a sentence satisfying the following conditions:

1. B is simple.
2. the syntax $\text{Syn}(B)$ is well ordered.

then B is called a base.

I take advantage of the fact that the syntax of a base is well ordered by representing the paths of it as r^k —which is to say $r^k \in \text{Syn}(B)$ —where r is its root (and only node), and k is the k^{th} path within the well ordering.

factorization of $\text{Syn}(U)$

In the case that $\text{Syn}(U)$ is factorizable, I will write:

3.3.1 an example

I’ll add an example for clarity (because examples are good):

The clause or formula g being informationally equivalent to:

$$(b + (cd))e$$

would be translated to have its syntax as:

$$g : 1 + \mu(1 + \sigma + \sigma^2(1 + \mu + \mu^2)) + \mu^2$$

which I must admit is comparatively long to convey the same structural information (about 3 times the length), but as with anything in language, frequently used “phrases” tend to evolve shorter forms; once the “whole” is established, the context is set and shortforms may be invoked, but I’ll get to that later.

In anycase, the components broken down are:

$$\begin{aligned} g_{/1} &= (b + (cd))e \\ g_{/\mu} &= b + (cd) \\ g_{/\mu^2} &= e \\ \\ g_{/\mu\sigma} &= b \\ g_{/\mu\sigma^2} &= cd \\ \\ g_{/\mu\sigma^2\mu} &= c \\ g_{/\mu\sigma^2\mu^2} &= d \end{aligned}$$

The strength does not entirely show itself here, because this formula is not informatically dense, but other examples with info-dense data environments (jungles) like “sums of sums of products” can better demonstrate.

The analogy is with algorithmics. Powerful algorithms that can do the same work as simpler ones tend to use more resources and are less efficient than their counterparts when computing small values; they only really show their true strength when heavier computations are required.

Finally, one may have noticed already I tend to use σ as the root of any formula of addition and μ as the root for any formula having to do with multiplication. I will use such conventions through the remainder.

Part II

Narrating
Enumerative Media

Chapter 4

Choices

4.1 Grammar

Choice

Choice Inventory

Cardinal Choice Inventory

Combination

Conjunctive Choice

Conjunctive Inventory

Disjunctive Choice

Disjunctive Inventory

Antipassive Choice

Reconciliation

Reconciliation Inventory

4.2 Declarations

All *sentences* referred to here are *grammatical* ones unless stated otherwise. For my exposition of *grammar theory* see my other article.

Here *sets* are left intuitive, for an exposition on axiomatic set theory refer to some other book. If X is some set then $|X|$ is its *cardinality*, if X is a finite set then I will refer to $|X|$ as its *size*. Given this to be a very common property, a more convenient notation is desired to say two sets have the same cardinality. Since this is the case, I will write $X \simeq Y$.

The declarations section of this and any other chapter is largely about defining various points of grammar. It is best to glance over it when reading the first time around; to get a better feel for what tools this book makes available to the user.

4.2.1 General Choice

Definition (Choice): Given the sentence

$$C : 1 + \diamond$$

if $C_{/1} \subseteq C_{/\diamond}$ then C is called a *choice* and is expressed in words as “ $C_{/1}$ is chosen from $C_{/\diamond}$.”

Definition (Choice Inventory): Let

$$\mathbf{C} : 1 + \diamond$$

be an operator of choices such that for any $C_1, C_2 \in \mathbf{C}$ and some set X we have

$$C_{1/\diamond} = X = C_{2/\diamond}$$

then \mathbf{C} is called a *choice inventory* over X .

Notice here the subtle uses of the subscript notation. Any label to the left of the first slash ‘/’ indicates a regular *subscript as label* in the traditional use of the notation. Here though the notations combine, which is possible since no ambiguity is introduced. One may think of it as similar to the *namespace* concept in the Xml markup language.

A choice inventory is said to be *full* if for every subset $A \subseteq X$ there is a corresponding choice $C \in \mathbf{C}$ with $C_{/\diamond} = A$. This is to say that \mathbf{C} is full if it contains every possible choice of its kind.

It is very common in practice to *modify* a choice or an inventory. In natural language this is equivalent to applying an adverb. Given these particular definitions, the only real option is to restrict the type of subsets being chosen. Such thinking leads to the following:

Definition (Cardinal Choice Inventory): Let \mathbf{C} be a choice inventory. For all $C_1, C_2 \in \mathbf{C}$, if $C_{1/1} \simeq C_{2/1}$, then \mathbf{C} is called a *cardinal choice inventory*.

One thing the reader might notice is the naming scheme of the definitions so far; “choice” is nice and simple, yet “cardinal choice inventory” is not so nice and simple. The general pattern is that the more elegant definition names are (ideally) left for the more important concepts. With that said, we may introduce the all important notion of a *combination*:

Definition (Combination): Let \mathbf{C} be a cardinal choice inventory over a finite set; in particular let $\mathbf{C}_{/\diamond} \simeq n$ and $\mathbf{C}_{/1} \simeq r$, then \mathbf{C} is said to be an *n choose r combination*. Such an inventory is denoted as ${}_n\mathbf{C}_r$.

In the special case where $r = 1$ we have what is called a *selection*.

4.2.2 Extending Choice

The next grammar points worth iterating here, are natural extensions of the concept of choice:

Definition (Conjunctive Choice): Given the sentence

$$C : 1 + \diamond + \diamond^2$$

If $C_{/1} \subseteq C_{/\diamond} \cup C_{/\diamond^2}$ then C is called a *conjunctive choice* and is expressed as “ $C_{/1}$ is chosen from $C_{/\diamond}$ and $C_{/\diamond^2}$.”

It’s worth mentioning that nothing is said regarding whether or not $C_{/\diamond} \cap C_{/\diamond^2} = \emptyset$. As with the general choice, we broaden this to an inventory:

Definition (Conjunctive Inventory): Let

$$\mathbf{C} : 1 + \diamond + \diamond^2$$

be an operator of conjunctive choices such that for any $C_1, C_2 \in \mathbf{C}$ and some sets X, Y we have

$$C_{1/\diamond} = X = C_{2/\diamond} \quad \text{and} \quad C_{1/\diamond^2} = Y = C_{2/\diamond^2}$$

then \mathbf{C} is called a *conjunctive inventory* over X and Y .

There is a special case of the conjunctive choice worth considering. It is when the subject is chosen from only one of the choice sets:

Definition (Disjunctive Choice): Given the sentence

$$C : 1 + \diamond + \diamond^2$$

If $C_{/1} \subseteq C_{/\diamond}$ or $C_{/1} \subseteq C_{/\diamond^2}$ but not both, then C is called a *disjunctive choice* and is expressed as “ $C_{/1}$ is chosen from $C_{/\diamond}$ or $C_{/\diamond^2}$.”

Definition (Disjunctive Inventory): Let

$$\mathbf{C} : 1 + \diamond + \diamond^2$$

be an operator of disjunctive choices such that for any $C_1, C_2 \in \mathbf{C}$ and some sets X, Y we have

$$C_{1/\diamond} = X = C_{2/\diamond} \quad \text{and} \quad C_{1/\diamond^2} = Y = C_{2/\diamond^2}$$

then \mathbf{C} is called a *disjunctive inventory* over X or Y .

Let $\mathbf{C} : 1 + \diamond + \diamond^2$ be an . If given any $W \subseteq \mathbf{C}_{/\diamond}$ there exists a unique choice $C \in \mathbf{C}$ with $C_{/1} = W$, then \mathbf{C} is called a *full combination*.

Let $\mathbf{C} : 1 + \diamond + \diamond^2$ be a disjunctive inventory with $\mathbf{C}_{/1}$ a singleton. is called a .

Does the commutative law hold?

4.2.3 Composing Choice

Before we can compose choice, we must first revisit general choice. By its very definition it is used in a passive way. This has to do with the general approach to set theory in that all things are declarative and not procedural. Keeping in line with this, rather than an *active* version of “choice” we take a *antipassive* strategy. This is to say, we indirectly describe an active form:

Definition (Antipassive Choice): Given the sentence

$$C : 1 + \diamond + \diamond^2$$

if $C_{/1} \subseteq C_{/\diamond^2}$ and $C_{/\diamond} \subseteq C_{/\diamond^2}$ with

$$C_{/1} \cap C_{/\diamond} = \emptyset \quad \text{and} \quad C_{/1} \cup C_{/\diamond} = C_{/\diamond^2}$$

then C is called a *antipassive choice* and is expressed in words as “ $C_{/1}$ chooses $C_{/\diamond}$ from $C_{/\diamond^2}$.”

Notice here that $C_{/1}$ and $C_{/\diamond}$ form a partition of $C_{/\diamond^2}$. The trick of words and logic is that by directly choosing $C_{/1}$ we are in fact indirectly choosing $C_{/\diamond}$ from $C_{/\diamond^2}$. It is in this form we are able to readily take compositions:

$$C' : 1 + \diamond + \diamond^2(1 + \diamond + \diamond^2)$$

4.2.4 Reconciling Choice

Definition (Reconciliation): A choice

$$\tilde{C} : 1 + \diamond$$

is said to be a *reconciliation* over \mathcal{H} if \mathcal{H} is an equivalence relation on $C_{/\diamond}$ such that $C_{/1}$ is one of its equivalence classes. It is read as: “ $C_{/1}$ is a reconciliation of $C_{/\diamond}$ over \mathcal{H} .”

Definition (Reconciliation Inventory): Let

$$\tilde{\mathbf{C}} : 1 + \diamond$$

be an operator of reconciliations all over some \mathcal{H} , then $\tilde{\mathbf{C}}$ is called a *reconciliation inventory* over \mathcal{H} .

Such general compositions are the motivation for a more direct process called *ranking* which is the subject of the next chapter. We now move on to counting theorems.

4.3 Enumerations

From a pedagogical point of view, it is a tempting strategy to resolve the issue of counting the value of the binomial coefficients by means of breaking the problem up into several lemmas. The thing is, the postulations in mind are worthy tools on their own and will thus be studied here with equal respect.

Theorem 4.3.1 (Binomial Coefficients) *Given counting numbers $n, r \in \mathbb{N}$, let ${}_n\mathbf{C}_r$ and ${}_n\mathbf{D}_r$ be combinations, then*

$${}_n\mathbf{C}_r \simeq {}_n\mathbf{D}_r$$

Proof Given $n \in \mathbb{N}$ and the definition of a combination, we have $\mathbf{C}_{/\diamond} \simeq n \simeq \mathbf{D}_{/\diamond}$. There necessarily exists a bijective mapping $\phi : \mathbf{C}_{/\diamond} \rightarrow \mathbf{D}_{/\diamond}$. Given any choice $C_1 \in \mathbf{C}$, we restrict ϕ to $C_{/1}$:

$$\phi|_{C_{/1}}$$

by way of bijection there is a coresponding (and unique) $D_{/1} \in \mathbf{D}$ with $\phi(C_{/1}) = D_{/1}$. Such correspondences define a bijective mapping from \mathbf{C} to \mathbf{D} , which is to say, a bijection between the combinations ${}_n\mathbf{C}_r$ and ${}_n\mathbf{D}_r$. ■

Notice the main arguement of this proof was that of showing “equality of size.” This is the most fundamental tactic of a combinatorialist, and so this reader is reminded the test consists of proving bijectivity, which is to say injectivity and surjectivity.

In light of this theorem, the following notation is the standard way to represent the common size to all ${}_nC_r$ combinations:

$$\binom{n}{r}$$

It belongs to an important class of numbers called *binomial coefficients*.

The next step would be to be able to numerically compute this value in an efficient way. As it turns out, it is best to start simpler when the actual counting is involved.

Theorem 4.3.2 (Sum Rule) *Let ${}_n\mathbf{A}_r$ and ${}_n\mathbf{B}_r$ be full. If ${}_n\mathbf{V}_r$ is full, then:*

$$|{}_n\mathbf{V}_r| = |{}_n\mathbf{A}_r| + |{}_n\mathbf{B}_r|$$

Proof

Chapter 5

Rankings

5.1 Grammar

Ranking Permutation Sifting

5.2 Declarations

5.2.1 General Ranking

Definition (Ranking): Given the sentence

$$R : 1 + \triangleleft$$

if $R_{/1}$ is an injective tuple over $R_{/\triangleleft}$ then R is called a *ranking* and is expressed in words as “ $R_{/1}$ is ranked from $R_{/\triangleleft}$.”

Definition (Permutation):

Definition (Sifting): Given a tuple \mathbf{x} and the sentence

$$S : 1 + \triangleleft$$

if $S_{/\triangleleft} = \mathbf{x}$ and $S_{/1} \subseteq \text{Ran}(\mathbf{x})$, then S is called a *sifting* of \mathbf{x} and is expressed in words as “ $S_{/1}$ is sifted from $S_{/\triangleleft}$.”

As stated in the previous chapter, the motivation for this definition is that of iteratively composing several antipassive choices. Given the antipassive choice

$$\mathring{C} : 1 + \diamond + \diamond^2$$

as one continues to compose it

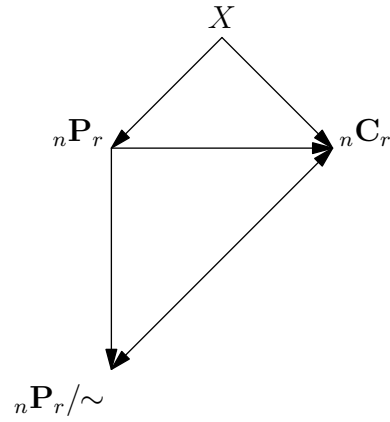
$$\mathring{C}' : 1 + \diamond + \diamond^2(1 + \diamond + \diamond^2)$$

“possibility space”... worthwhile definition?

Theorem 5.2.1 (Binomial Coefficients Again) *Given counting numbers $n, r \in \mathbb{N}$, we have*

$$\binom{n}{r} = \frac{n^r}{r!}$$

Proof The proof follows by analysing the process of choice. If we wish to choose k elements from n , The simplest way to conceptually construct a translation of mappings to evaluate size of the possibility space is to choose one element at a time. When this happens though, we are left with an “n rank r” permutation. Where do we go from here? we will have to deconstruct the situation.



Things begin falling together when we observe that for our interests, a different ranking of the same elements has no consequence on our final choice. This is to say these rankings for our purposes are equivalent. We may thus build an equivalence relation using a sifting criterion to determine equivalence. Extending this, we consider all possible rankings that would result in the same end-choice when sifted.

We should frame things then in the context of a reconciliation. When we do this we broaden our perspective and look at the full permutation inventory. From this we form the reconciliation inventory for we would have a bijection from this inventory to the combination inventory of interest. ■

Part III

Deconstructing Enumerative Media

Part II focused on introducing the idea of choice and exploring its non-problematic extensions. This part begins to analyse the variety of choices which produce problematic enumerations.

In the previous part, both the syntax and semantics of grammar were of a tree-sentence nature. We begin to decouple this pairing by looking at enumerations which maintain tree-structure syntax but produce non-tree semantics. This is to say there begins to be some overlap.

This effects the enumeration, especially in its representation which must now parse a more general graph-structure.

Chapter 6

Inclusion-Exclusion

In literary analysis, the notion of *identity* is very much intertwined with the notion of an *insider*. Just as well, and unfortunately it being more readily or easily defined is the notion of an *outsider*.

In a pluralistic or collectivistic world though, one's identity often more complicated than a simple duality; it generally consists of being included in some groups, and excluded from others. In the context of set-theory, we would say, given some object x that some set X is an *in-set* if $x \in X$ and otherwise would be called an *out-set* of x .

The problematics of plurality I might return to with the concept of a color-model *gamut*.

In the previous chapter our definition of a ranking was motivated by the need to (for a current lack of a better way to put it) economize the process of repeated choice generated from non-overlapping sets. This chapter then could be said to be motivated by the need to economize the same process but this time being generated from overlapping sets.

If we were to translate the syntax of the semantics itself, I suspect in this case it would look like a bipartite graph.

Chapter 7

Pigeonholes

The analysis of inclusion-exclusion was the start-off in breaking down our assumptions when enumerating cardinality. The choice of it as the start-off was strategic in that it still is intimately tied in with choice as our enumerative media.

In this chapter it is the intention to now make a clear break from “choice” and analysis another collection of structures that will aid us in understanding what assumptions we’ve been using thus far in our communications.

A pigeonhole is a situation demonstrating the limitations of choice and such enumerative consequences. Intuitively it arises when one has to make more choices than there are objects to be chosen. The limitation is one of *distribution* and comes in realizing that any given choice inventory will have at least one pair of non-disjoint choices.

Part IV

Reconstructing Enumerative Media

In this part, we begin to look at alternative frameworks for enumerating the various inventories of interest. We are now ready to look a broader graph-clause structures as means of enumeration.

Chapter 8

Generating Structors

“Structor Grammar?”

We begin to look at enumeration from a global perspective.

8.1 Formal Power Series

8.2 Generating Functions

The first thing to note is that such a syntax $\text{Syn}(B)$ is finite, which holds since we are intending to define polynomials which are themselves finite in degree within rings. Using the notation defined as above, the paths of a base are represented as x^0, x^1, \dots, x^n for some finite n , where x is the root. I should add I’m using the natural numbers \mathbb{N} to “notate” the well ordering.

To get the big picture of “all at once” though, I *concatenate* these paths using the plus (+) sign:

$$\text{Syn}(B) := 1 + x + x^2 + \dots + x^n$$

Now I cheated a little here, so I will explain: the ‘1’ is shorthand for x^0 , as well, the x is shorthand for x^1 . Also, as stated before, if I want to display any of the arguments of B , I simply write them as:

$$B/x^k \quad \text{or} \quad x^k \setminus B$$

where k is some natural number from 0 to n .

With this I have enough for our definition.

Let R be some ring, we wish to define a collection of bases \mathcal{B} possessing the same cardinality (size) as R such that they all contain the same syntax, one with the following form:

$$\text{Syn}(B) := 1 + x + x^2 + \dots + x^n$$

for $B \in \mathcal{B}$ and some fixed n . In particular define their arguments as:

$$B/x^k = a^k \quad \text{for some } a \in R \text{ with } 0 \leq k \leq n$$

and do this for each $B \in \mathcal{B}$ such that they span all a in R .

This is $\frac{1}{2}$ of the picture though, we still need coefficients. How does one obtain these you ask?...through algebra’s favorite trick...an equivalence relation of course.

The thing to note is that the bases $B \in \mathcal{B}$ possess mappings $m : \text{Dom}(B) \rightarrow R$ all with the same range, so given some other mapping $g : R \rightarrow R$, one may “act” on these bases by defining compositions:

$$g \circ m : \text{Dom}(B) \rightarrow R$$

Thus we define an “action map” to create our equivalence relation, where our definition of two bases $B, B' \in \mathcal{B}$ being equivalent is:

$$B \equiv B' \iff g \circ m = g \circ m'$$

where m is the map of B and m' is the map of B' . In such a case, to represent an equivalence class, one writes:

$$[B] = c_0 + c_1x + c_2x^2 + \dots + c_nx^n$$

where the c_k are called coefficients, possessing the following property:

$$c_k = g([B]/x^k) \quad \text{given } 0 \leq k \leq n$$

This is possible because each base in a class has by definition the same syntax and the same coefficients. In such a case this equivalence class is called a polynomial.

Finally, if one has many such polynomials of various “degrees”, it is easy then to define addition and multiplication to obtain a polynomial ring $R[x]$... and the rest they say is history.

8.3 Reencoding Choice

8.4 The Binomial Theorem

$$(x + y)^n = \sum_{0 \leq r \leq n} \binom{n}{r} x^{n-r} y^r$$

Chapter 9

Recurrence Relations