Notes on an Application of Combinatorial Compositions, Nexus Numbers, and the Divisor Function

Andrei Lapets

July 05, 2005

1 Overview

Our goal is to solve a counting problem. Given some number $n \in \mathbb{N}$ which has a prime factorization

$$n = p_1^{\alpha_1} \cdots p_r^{\alpha_r},$$

we want to determine the number $\delta_k(n)$ of unique, finite ordered tuples of the form

$$\langle m_1, \ldots, m_k \rangle$$

such that $m_1 \cdots m_k = n$ and $\forall i,\ m_i \neq 1$. Note that the tuples must be unique, which means permutations of multiple occurrences of some value m_i in the tuple should not contribute to the overall count. Note that by the structure theorem for abelian groups, this count is equivalent to the number of distinct finite abelian groups of order n up to isomorphism. We will investigate a number of approaches for solving this problem.

2 Generating Functions

We will first attempt to find an intuitive interpretation using generating functions for both squarefree and non-squarefree numbers. We would also like to find a closed form for $\delta_k(n)$ for non-squarefree numbers n. In order for this approach to work, we need to describe an equivalent counting problem.

Our analogous counting problem involves the placement of a certain number of distinguishable, distinct items into a certain collection of *labelled* bins where order inside the bins does not matter. One example of this might be the separation of a number composed of single instances of distinct prime factors; in other words, some $c \in \mathbb{N}$ such that $c = p_0 \times p_1 \times \ldots \times p_k$ and $\forall$ $0 \leq i, j \leq k$, $p_i \neq p_j$. In most cases, a simple combinatorial composition will suffice to solve this counting problem. However, there exists a possible *additional* constraint which can make this problem more difficult: the necessity to ignore empty bins (or, in our example, bins that contain the 1 factor) when considering unique arrangements. This would allow us to peruse the collection in each bin in some order, and completely ignore any empty bins. A unique arrangement in this case would be a result of such a perusal. In order to count the number of possible unique arrangements given some number of bins and items we must completely ignore the bins which are empty to avoid counting certain identical arrangements multiple times. This is the most difficult aspect of the simplified, squarefree version of our problem.

2.1 Combinatorial Compositions

Let us describe the problem by assuming that there are k labelled bins and n unique items. We want to define a function for the number of ways we can arrange these n items across k bins such that order inside the bins does not matter and empty bins are ignored when comparing any two arrangements.

We will use the standard definition of combinatorial compositions. First, let us show how this is conceptually related to our problem. In the case of each bin, we can employ an ordinary generating function for each item.

$$\frac{1}{1-x} = 1 + x + x^2 + x^3 + \ldots$$

Note that each contributes once. Thus, for the number of items n, we should be able to determine how many ways we can utilize it using $(\frac{1}{1-x})^n$. Note that we need not ensure that it is used at least once, because we need to ensure that every position is utilized over all the prime factors, not just a single one. In addition, this requires us to supply another $(\frac{1}{1-x})$ factor to our product, because we need to account for the fact that 1 factors must be used where a certain p_i factor is not used (or, in other words, that p_i^0, instead of p_i^j for some j, is used). This leads us to an ordinary generating function which has a closed form.

$$\left(\frac{1}{1-x}\right)^{n+1}$$

We know from the binomial theorem that the k^{th} coefficient in the above generating function takes the form $\binom{n+k-1}{k}$. We know there are k possible distinct positions to utilize, and that any given number of prime factors that can be utilized is n, which leads to our $n+1$ exponent as described above. Thus, our desired count for the set of n items will be $\binom{n+1+k-1}{k} = \binom{n+k}{k}$. Note that this is effectively the multiplicative equivalent of the traditional formula for combinatorial compositions (the number of compositions of $n+1$ into k parts or factors, which intuitively makes sense in this case, as does the fact that the 1 is added to account for the multiplicative identity), which likewise utilizes the multiplicative identity 1 in place of the additive identity 0. This will cause a problem for us shortly. Our notation in this case will be $C_k(n+1) = \binom{n+1+k-1}{k}$.

First, we define a function $\delta_n(k)$ for the number of ways we can arrange a system using k positions. We will need to provide a recursive definition for the moment.

$$C_k(n+1)$$

Naturally, because we want to avoid counting examples where not all k positions are utilized, we will need to subtract from this. We note that the number of ways an arrangement utilizing $k-1$ positions can occur over k positions is exactly k (in other words, the 1 which we want to avoid can be in one of k possible positions). This leads to our recursive definition.

$$\delta_n(k) = \Big(C_k(n+1)\Big) - k\delta_n(k-1)$$

We note that we can write the solution to this relation explicitly:

$$\delta_n(k) = C_k(n+1) - \sum_{i=0}^{k-1}\Big[(-1)^i\Big(\prod_{j=0}^{i} k-j\Big)C_{k-i-1}(n+1)\Big].$$

This can be simplified.

$$\delta_n(k) = C_k(n+1) - \sum_{i=0}^{k-1}\Big(\frac{(-1)^i k!}{(k-i-1)!}C_{k-i-1}(n+1)\Big)$$

We note that the above result is cumbersome and difficult to follow, and is of little value when we are faced with a given problem, though it is sufficient to calculate our answer. We can analyze our function δ_n with the help of this equation to find a more suitable method to solve our problem.

3 Nexus Numbers

Analysis of the patterns in this situation can lead to a more elegant method for counting the number of unique arrangements.

Note the sequences for a fixed k but a varying value of i for the first few k.

$$\{\delta_i(2) \mid i \in N,\ i \geq 1\} = \{1, 2, 3, 4, 5, ...\}$$

We can extend this process indefinitely to find the equation for $\delta_n(k)$ for some $k \in \mathbb{N}$.

$$\delta_n(7) = (7^n - 6^n) - 5(6^n - 5^n) + 11(5^n - 4^n) - 13(4^n - 3^n) + 9(3^n - 2^n) - 3(2^n - 1^n) + 1$$
$$\delta_n(6) = (6^n - 5^n) + 4(5^n - 4^n) - 7(4^n - 3^n) + 6(3^n - 2^n) + 3(2^n - 1^n)$$

$$\delta_n(5) = (5^n - 4^n) - 7(4^n - 3^n) + 6(3^n - 2^n) + 3(2^n - 1^n)$$
$$\delta_n(4) = (4^n - 3^n) + 6(3^n - 2^n) + 3(2^n - 1^n)$$
$$\delta_n(3) = (3^n - 2^n) - (2^n - 1^n) + 1$$
$$\delta_n(2) = (2^n - 1^n)$$
$$\delta_n(1) = 1$$

The first thing to note about these equations is that the terms with exponents are, in face, nexus numbers of the form $i^n - (i-1)^n$ [Weisstein 2] for $i \epsilon \mathbb{N}$ such that $0 < i \leq k$.

$$N_{n-1}(i-1) = (i)^n - (i-1)^n$$

This means every term can be expressed as a polynomial.

$$N_{n-1}(i-1) = \sum_{j=0}^{n-1} \binom{n}{j} (i-1)^j$$

Likewise, we see that the coefficients of the terms in each case are actually a set of triangle numbers. The numbers here have a corresponding formula [Sloane, A108561], which is defined recursively.

$$T(n,0) = 1$$
$$T(n,n) = (-1)^{\lfloor \frac{n}{2} \rfloor}$$

The remaining terms can be obtained using the recursive formula for some n and k such that $0 < k < n$.

$$T(n+1,k) = T(n,k-1) + T(n,k)$$

Luckily, the coefficients for each nexus term also have an explicit formula.

$$\chi_m(3) = \sum_{i=0}^{m} (-1)^{(m-i)} \binom{i+3}{3}$$

The $n = 3$ [Sloane, A002623] and $n = 4$ [Sloane, A001752] cases are available, and are examples of the more general formulation, which we will call $\chi : \mathbb{Z} \to \mathbb{Z}$, and define as follow:

$$\chi_m(k) = \sum_{i=0}^{m} (-1)^{(m-i)} \binom{i+k}{k}$$

We will rewrite the above in a form more suited for our purposes, allowing for shorter parameters. In this form, k represents the enumeration of the nexus number term for which we need the coefficient, and m represents the number of bins.

$$\chi_m(k) = \sum_{i=0}^{m-1-k} (-1)^{(m-1-k-i)} \binom{i+k}{k}$$

We now see that we can easily write a general form for $\delta_n(k)$ using χ and the nexus numbers.

$$\delta_n(k) = \sum_{j=0}^{k-1} \Big[(-1)^{k-j+1} \; \chi_k(j) \; N_{n-1}(j) \; \Big] \tag{1}$$

We provide the expanded version for clarity.

$$\delta_n(k) = \sum_{j=0}^{k-1} \Big[(-1)^{k-j+1} \Big[\sum_{i=0}^{n} (-1)^{(n-i)} \binom{i+j}{j} \Big] \times \Big[\sum_{i=0}^{n-1} \binom{n}{j} (n-1)^i \Big] \Big] \tag{2}$$

We have now derived an explicit formula for the case where there is only a single set N of n distinguishable items.

3.1 The General Case

We will now consider a union S of r disjoint sets:

$$S = \bigcup_{i=1}^{r} S_i \; ,$$

$$S_i \cap S_j = \emptyset \; \forall i, \; j \; , \; 0 \; < \; i, \; j \; \leq \; r, \; i \neq j \; .$$

We can imagine each set S_i containing some number of items of a certain type, and the number of sets r being the number of unique types. We now wish to once again count the number of ways in which these items can be placed into k labelled bins. To review, order in each bin does not matter, and empty bins are skipped when considering a unique arrangement.

We utilize a concise representation of the size of each set S_i for clarity:

$$s_i = |S_i|.$$

We once again remind ourselves that we want to find some $\delta_S(k)$ for a given set S and some number of labelled bins k.

3.2 Indistinguishable Items

Let us first consider the case of where $r = 1$. In this case, we have some set $S_\perp$ such that $s_\perp = |S_\perp|$. The function $\delta_{S_\perp}$ is easily derived:

$$\delta_{S_\perp}(k) = \sum_{j=0}^{\lfloor \frac{(k-1)}{2} \rfloor} \binom{s_\perp}{k-1-2j}.$$

We note that this initial form will be the necessary first base case in any calculation of $\delta_S(k)$.

3.3 Form of $\delta_S(k)$

We now wish to find a general form for δ which utilizes χ and the nexus numbers. This will require that we rebuild our set of equations $\delta_m(k)$ for various values of k above into recursive equations. At each step of the recursion, we will factor the new combinations into the equation. For any set S where for all i, $|S_i| = 1$, the recurrence relation should be equivalent to our explicit formulation (1).

The recurrence relation for $\delta_n(k)$, can be described as follows:

$$\delta_1(k) = 1 \ , \tag{3}$$

$$\delta_{n+1}(k) = \sum_{i=0}^{k-1} \delta_n(k) \ . \tag{4}$$

We can now sprinkle the combination factors across this definition to account for the fact that at each step, we may have a different subset $S_i \subset S$. We need a base case for our recurrence:

$$\delta_{S_\perp}(k) = \sum_{j=0}^{\lfloor \frac{(k-1)}{2} \rfloor} \binom{s_\perp}{k-1-2j} \ .$$

We now form the recurrence using the observation that $\delta_{(S \bigcup S')}(k)$ can be found given some $\delta_S(k)$ by multiplying the previous iteration of the recurrence by the size of the maximum term inside a nexus number, which is equivalent to the number of bins. This means we will have a summation of k terms. We then need to ensure that we remove the new terms which result from the fact that while $k(k^n) = k^{n+1}$, $k((k-c)^n) = (k-c)^{n+1} + \varepsilon$, where in this case $\varepsilon = (c)(k-c)^n$. Let $u(S)$ denote the number of unique types of items already contained in S:

$$\delta_{(S \bigcup S')}(k) = \sum_{i=0}^{k-1} \Big[\binom{|S'| + i - 1}{i} \Big[\delta_S(k) - \sum_{j=0}^{i} \Big[(-1)^{k+j-1} (j^{u(S)}) (\chi_k(j) + \chi_k(j-1)) \Big] \Big] \Big] \ .$$

We can now find the unique number of arrangements in the problem we described by simply calculating $\delta_S(k)$ for a specific number of labelled bins k and set S. Note that we have not shown the equivalence of the first explicit formula for $\delta_S(k)$ to the formula utilizing the nexus numbers, and that the equivalence of the two expression in the general case remains to be proven.

4 The Divisor Function

We note that the divisor function σ_0, defined by

$$d(n) = \sigma_0(n) = \sum_{d|n} d^0 = \sum_{d|n} 1,$$

can offer us the most straightforward way to perform the desired count. We first define the iterated s-fold divisor function as

$$d_s(n) = \underbrace{\sum_{d_s|n} \sum_{d_{s-1}|d_{s-2}} \dots \sum_{d_1|d_2}}_{s} 1.$$

Note that the above function yields the number of ordered s-tuples of the form $\langle d_1, \dots, d_s \rangle$ where

$$1|d_1|d_2|\dots|d_s|n.$$

We can now set $d_0 = 1$, $d_{s+1} = n$, and

$$s = k - 1,$$

to obtain $\langle d_0, d_1, \dots, d_k \rangle$ where

$$d_0|d_1|\dots|d_{k-1}|d_k.$$

Note that we can use one of the above tuples to rewrite the elements of our desired tuples $\langle m_1, \dots, m_r \rangle$ for $1 \leq i \leq k$:

$$m_i = \frac{d_i}{d_{i-1}}.$$

There is a problem here, because $1|n$ and $n|n$, and we required that $\forall i, m_i \neq 1$. We can account for this by subtracting 2 at each level of the definition for $d_k(n)$ except the last:

$$d_s(n) = \sum_{d_s|n} \Bigg(\sum_{d_{s-1}|d_{s-2}} \Bigg(\cdots \Bigg(\sum_{d_1|d_2} 1 \Bigg) - 2 \ldots \Bigg) - 2 \Bigg) - 2,$$

which can be rearranged, once again with $s = k - 1$, as

$$\delta_k(n) = d_{k-1}(n) - 2 \sum_{i=1}^{k-2} d_i(n).$$

Note that for any $k > d(n)$, it is the case that $\delta_k(n) = 0$. Thus, we can define δ in general as

$$\delta(n) = \sum_{i=1}^{\infty} \delta_i(n)$$

Because we assume we already know the prime factorization of n, this calculation is no more and no less complex than any of the previous formulas. If the number of factors of n is $d(n)$, we can bound the complexity of the calculation by the approximation $O(d(n)^k)$.