#A89 INTEGERS 18 (2018)

THE COMBINATORICS OF EVENLY SPACED BINOMIAL COEFFICIENTS

Nicholas A. Loehr[1]
Dept. of Mathematics, Virginia Tech, Blacksburg, Virginia
nloehr@vt.edu

T. S. Michael
Dept. of Mathematics, U. S. Naval Academy, Annapolis, Maryland

Received: 2/6/18, Accepted: 10/21/18, Published: 10/26/18

Abstract

A curious identity for binomial coefficients states that

$$\sum_k \binom{n}{km} = \frac{1}{m}\sum_{j=0}^{m-1}(1+e^{2\pi ij/m})^n.$$

There are similar formulas for the sum of $\binom{n}{a}$ over all a's with a given remainder mod m. This paper undertakes a combinatorial exploration of these formulas emphasizing bijective proofs. We derive multivariate generating functions for these sums using both linear-algebraic arguments and sign-reversing involutions. Next we offer a combinatorial explanation of why these sums are "almost" $2^n/m$. We give a bijective proof that the minimum of the sums $\sum_k \binom{n}{km+r}$ equals $(2^n-\ell(n,m))/m$, where the "error term" $\ell(n,m)$ has an explicit combinatorial interpretation involving words satisfying certain parenthesis-matching conditions. Among other consequences, this leads to a novel combinatorial model for alternate Lucas numbers.

Dedication. This paper is dedicated to my late coauthor T. S. Michael, a wonderful friend, collaborator, teacher, and mathematician, whose untimely death in 2016 saddened us all.

[1] This work was partially supported by a grant from the Simons Foundation (#244398 to Nicholas Loehr).

1. Introduction

The following binomial coefficient identities are very well known:

$$\sum_{k\ge 0}\binom{n}{k} = 2^n \text{ for } n \ge 0; \qquad \sum_{k\ge 0}\binom{n}{2k} = \sum_{k\ge 0}\binom{n}{2k+1} = 2^{n-1} \text{ for } n \ge 1.$$

A less widely known formula gives the sum of evenly spaced binomial coefficients in terms of a complex root of unity: given integers r, m, n with $0 \le r < m$, we have

$$\sum_{k\ge 0}\binom{n}{km+r} = \frac{1}{m}\sum_{j=0}^{m-1}\omega^{-jr}(1+\omega^j)^n, \text{ where } \omega = e^{2\pi i/m}. \tag{1}$$

This formula is stated in Gould's encyclopedic compendium of binomial coefficient identities [4, Formula (1.53)], and a special case is given as Problem 1.42(f) in Lovász's book [9].

Formula (1) can be proved algebraically by expanding the right side using the Binomial Theorem and using the fact that $\sum_{i=0}^{m-1}(\omega^j)^i$ is m if m divides j and is 0 otherwise. Guichard [5] gives the details of this derivation and proves the following trigonometric version of the formula (also stated in [4, Formula (1.54)]):

$$\sum_{k\ge 0}\binom{n}{km+r} = \frac{2^n}{m}\left[1 + 2\sum_{k=1}^{\lfloor (m-1)/2\rfloor}\cos^n\frac{\pi k}{m}\cos\frac{\pi k(n-2r)}{m}\right]. \tag{2}$$

Benjamin, Chen, and Kindred [1] gave a *combinatorial* proof of (1) by showing that both sides count walks in a cycle graph with loops, where each walk is weighted by a certain power of ω.

For small values of m, the right side of (1) can be simplified to expressions that do not involve complex numbers or cosines. For example,

$$\sum_{k\ge 0}\binom{n}{3k} = \frac{2^n + \epsilon(n)}{3},$$

where $\epsilon(n) \in \{\pm 1, \pm 2\}$ depends on n mod 6. Benjamin and Scott [2] gave combinatorial proofs of this formula and a similar formula for $m = 4$. For $m = 5$ and $m = 10$, Howard and Witt [6] obtained formulas involving Fibonacci numbers, Lucas numbers, and Pell numbers by algebraic methods. Results based on the combinatorial trace method appear in [3, 8].

This paper further explores the rich combinatorics of the sums $S(n, m, r) = \sum_{k\ge 0}\binom{n}{km+r}$ using a variety of tools: generating functions, deterministic finite automata (DFAs), linear algebra, recursions, involutions, and bijections. Our initial goal (proposed by my late coauthor T. S. Michael in 2015) was to find *bijective* proofs of identities involving the sums $S(n, m, r)$. However, since the right side of (1) is an expression involving the complex quantity ω, one must first clarify what one even means by a bijective proof in this situation. We give several answers to this question in this paper.

Our first answer involves a concise generating function for the sums $S(n,m,r)$:

$$\sum_{n\geq 0} S(n,m,r)x^n = \frac{x^r(1-x)^{m-1-r}}{(1-x)^m - x^m} \quad \text{for } 0 \leq r < m. \tag{3}$$

Initially, we derive this generating function by analyzing a DFA and using Cramer's Rule to solve a nearly-diagonal linear system. Then we provide a fully bijective proof of (3) by defining a sign-reversing, weight-preserving involution on a certain set of objects. Taking the partial fraction expansion of (3) leads back to the original summation formula (1). We also give some multivariate extensions of this generating function, including a version for q-binomial coefficients.

The next part of the paper provides a bijective explanation of the fact that $S(n,m,r)$ is "almost" equal to $2^n/m$. Let $W_n = \{0,1\}^n$ denote the set of binary strings of length n. For each $w \in W_n$, let $N_1(w)$ (resp. $N_0(w)$) be the number of 1's (resp. 0's) in w. For each fixed $m > 0$, W_n can be written as the union of m pairwise disjoint sets $W_{n,m,r}$ for $0 \leq r < m$, where

$$W_{n,m,r} = \{w \in \{0,1\}^n : N_1(w) \equiv r \pmod{m}\}.$$

Since $\binom{n}{b}$ counts binary strings $w \in \{0,1\}^n$ with $N_1(w) = b$, it is clear that $S(n,m,r) = |W_{n,m,r}|$. We would like to find bijections between the sets $W_{n,m,r}$ (as r varies), which would show that all of these sets have size $2^n/m$. For instance, we could look for a bijection $f : W_n \to W_n$ such that for all $w \in \{0,1\}^n$,

$$N_1(f(w)) \equiv N_1(w) + 1 \pmod{m} \qquad \text{and } f^m(w) = w. \tag{4}$$

Of course, such maps f cannot possibly exist for general m, since $2^n/m$ is not even an integer!

We modify this initial idea by restricting the domain and codomain of f to some subset of W_n consisting of "good" binary strings. For example, consider $n = 7$ and $m = 5$. By a variation of Pascal's Triangle (see §3.1), we can quickly compute

$$S(7,5,0) = 22,\ S(7,5,1) = 14,\ S(7,5,2) = 22,\ S(7,5,3) = 35,\ S(7,5,4) = 35.$$

Certain symmetries in this list (i.e., two sets of size 22 and two sets of size 35) are easily explained. We are more interested in the *minimum* value of $S(n,m,r)$ as r varies; in this case, the minimum value is 14. Any bijection f having the properties described in the last paragraph can have at most 14 5-cycles, since each such cycle visits a different element of $W_{7,5,1}$. Thus, the best possible bijection of this form could only visit 14 objects from each remainder category $W_{7,5,r}$, leaving 8 unused words in $W_{7,5,0}$, 0 unused words in $W_{7,5,1}$, 8 unused words in $W_{7,5,2}$, 21 unused words in $W_{7,5,3}$, 21 unused words in $W_{7,5,4}$, and 58 total leftover words in $\{0,1\}^7$.

For general n and m, the "bottleneck" in the bijection is caused by the *minimum sum*

$$\mu(n,m) = \min_{r:\ 0\leq r<m} S(n,m,r). \tag{5}$$

Define the total number of *leftover* input objects to be

$$\ell(n,m) = 2^n - m \cdot \mu(n,m). \tag{6}$$

For fixed n and m, an *optimal* bijection of the form proposed above would consist of a set $D \subseteq \{0,1\}^n$ and a map $f : D \to D$ satisfying (4) for all w in D, together with a decomposition of $\{0,1\}^n$ into pairwise disjoint sets $A_0, A_1, \ldots, A_{m-1}, L$ such that: $A_r \subseteq W_{n,m,r}$ for $0 \leq r < m$, $|A_r| = \mu(n,m)$ for $0 \leq r < m$, $|L| = \ell(n,m)$, and $D = \bigcup_{r=0}^{m-1} A_r$. The conditions on $|A_r|$ and $|L|$ are equivalent to requiring that for some r, A_r is the entire set $W_{n,m,r}$. Rearranging (6), we see that the map f provides a *bijective* proof of the formula

$$\min_{r:\ 0\leq r<m} \sum_{k\geq 0} \binom{n}{km+r} = \frac{2^n - \ell(n,m)}{m}. \tag{7}$$

We construct such bijections in this paper for all n and m and provide explicit combinatorial interpretations for the sets A_r and L. In particular, we will see that $\ell(n,m)$ counts strings of left and right parentheses of length n such that every prefix has at most $m-2$ unmatched parentheses within that prefix. For small values of m, the numbers $\ell(n,m)$ have predictable structure. For example:

- $\ell(n,2) = 0$ for all $n > 0$;
- $\ell(n,3)$ is 1 for n even and 2 for n odd;
- $\ell(n,4) = 2^{1+\lfloor n/2 \rfloor}$ for $n > 0$;
- $\ell(n,5) = \mathrm{Luc}_{n+1}$ for n even and $2\,\mathrm{Luc}_n$ for n odd, where Luc_i is the ith *Lucas number* (defined recursively by $\mathrm{Luc}_0 = 2$, $\mathrm{Luc}_1 = 1$, $\mathrm{Luc}_n = \mathrm{Luc}_{n-1} + \mathrm{Luc}_{n-2}$ for $n \geq 2$).

The special cases $m = 3$ and $m = 4$ yield bijective evaluations of $S(n,3,r)$ and $S(n,4,r)$ that are likely equivalent (up to notation translation) to the proofs given in [2]. The $m = 5$ result provides a novel combinatorial interpretation for alternate Lucas numbers in terms of parenthesis-matching conditions. For larger values of m, the numbers $\ell(n,m)$ are less predictable. But, the generating functions for these numbers can be computed by simple recursive formulas. We derive these recursions by analyzing more deterministic finite automata.

The rest of this paper is organized as follows. Section 2 gives linear-algebraic and bijective proofs of the generating function (3) and its generalizations. Section 3 constructs the optimal bijections proving (7). Section 4 studies the *leftover numbers* $\ell(n,m)$ more closely, obtaining explicit trigonometric formulas and recursions for the associated generating functions. We conclude by proving the unexpected formula for $\ell(n,5)$ involving the odd-indexed Lucas numbers.

2. Generating Functions for $S(n,m,r)$

This section gives a linear-algebraic proof and a bijective proof of the following two-variable version of the generating function (3). Recall that $W_{n,m,r}$ is the set of words w in $\{0,1\}^n$ where the number of 1's in w is congruent to r modulo m, and $S(n,m,r)=|W_{n,m,r}|$.

Theorem 1. *For all $m>0$ and $0\leq r<m$, we have*

$$G_{m,r}(y,z) := \sum_{n=0}^{\infty}\sum_{w\in W_{n,m,r}} y^{N_1(w)}z^{N_0(w)} = \frac{y^r(1-z)^{m-1-r}}{(1-z)^m-y^m}. \tag{8}$$

Since $N_1(w)+N_0(w)=n$ for all $w\in\{0,1\}^n$, specializing $y=x$ and $z=x$ in (8) produces (3).

2.1. Proof via Finite Automata and Linear Algebra

Fix $m>0$ and $0\leq r<m$. It turns out that the set of strings $W_{*,m,r}=\bigcup_{n\geq 0}W_{n,m,r}$ is a *regular language* that can be recognized by a *deterministic finite automaton* (DFA). We refer the reader to Sipser's textbook [10] for the definition of finite automata and related notation used here.

The following DFA, denoted $\mathrm{DFA}_{m,r}$, recognizes the language $W_{*,m,r}$. The set of states is $\{0,1,\ldots,m-1\}$. State 0 is the start state, and state r is the only accepting state. For $0\leq i<m-1$, there is a transition from state i to state $i+1$ labeled by the symbol 1. There is also a transition from state $m-1$ to state 0 labeled by 1. For $0\leq i<m$, there is a transition from state i to itself labeled by 0. Figure 1 illustrates this DFA in the particular case $m=5$, $r=2$.

Each binary word $w\in\{0,1\}^*$ is processed by the DFA as follows. We begin in the start state. We read each symbol in w and follow the appropriate transition to a new state. After reading all symbols of w, the DFA *accepts* w if and only if our current state is an accepting state. It is routine to check that the set of strings accepted by $\mathrm{DFA}_{m,r}$ is precisely $W_{*,m,r}$. The key point is that at every stage of processing, we are in state i if and only if the number of 1's seen so far is congruent to i mod m. Thus, for every i between 0 and $m-1$, the generating function for the

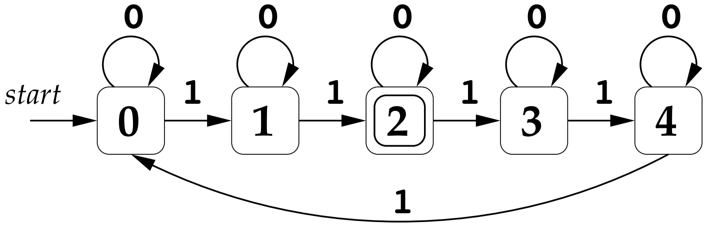

Figure 1: DFA recognizing the language $W_{*,5,2}$.

words w that end in state i of $\text{DFA}_{m,r}$ is $G_{m,i}(y,z)$. For brevity (recall m is fixed), we abbreviate $G_{m,i}(y,z)$ as G_i in the following discussion.

The m unknown formal power series $G_0, G_1, \ldots, G_{m-1}$ are related by m linear equations coming from the DFA. Consider state 0, for example. To reach state 0 at the end of the computation, we either append the symbol 0 to a word counted by G_0, or append the symbol 1 to a word counted by G_{m-1}, or use the empty word (since state 0 is the start state). Since each 0 is weighted by z and each 1 is weighted by y, the previous sentence is encoded by the generating function equation $G_0 = zG_0 + yG_{m-1} + 1$. Similarly, for each state i with $0 < i < m$, we follow transitions in the DFA to get $G_i = zG_i + yG_{i-1}$. These m equations constitute a linear system $A\mathbf{x} = \mathbf{b}$, where $\mathbf{x}$ is the column vector of G_i's, $\mathbf{b} = \mathbf{e}_0$ has a 1 in position 0 and zeroes elsewhere, and A is a circulant matrix with entries $1-z$ on the main diagonal and $-y$ on the next lower diagonal (as well as the $0,(m-1)$-entry). For example, when $m = 5$ and $r = 2$, the linear system is:

$$\begin{bmatrix} 1-z & 0 & 0 & 0 & -y \\ -y & 1-z & 0 & 0 & 0 \\ 0 & -y & 1-z & 0 & 0 \\ 0 & 0 & -y & 1-z & 0 \\ 0 & 0 & 0 & -y & 1-z \end{bmatrix} \begin{bmatrix} G_0 \\ G_1 \\ G_2 \\ G_3 \\ G_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{9}$$

Since state r is the only accepting state, we must solve for G_r in this linear system. (In fact, varying r does not change the system, so we really need to find all the unknowns $G_0, \ldots, G_{m-1}$.) Evidently, the column vector of G_i's is the first column in the inverse of the coefficient matrix A. We can quickly compute each G_i using Cramer's Rule for solving linear systems: $G_i = \det(A[i])/\det(A)$, where $A[i]$ is the matrix obtained from A by replacing column i by the right side $\mathbf{e}_0$ (indexing rows and columns starting at 0). Expanding the determinant along the first row, we see that $\det(A) = (1-z)^m - y^m$. On the other hand, expanding $A[r]$ along column r, we get $\det(A[r]) = y^r(1-z)^{m-1-r}$. For instance, when $m = 5$ and $r = 2$, computing $\det(A[2])$ as indicated leads to the 4×4 block-diagonal determinant

$$\det \begin{bmatrix} -y & 1-z & 0 & 0 \\ 0 & -y & 0 & 0 \\ 0 & 0 & 1-z & 0 \\ 0 & 0 & -y & 1-z \end{bmatrix} = y^2(1-z)^{5-1-2}.$$

By Cramer's Rule, we deduce $G_r = y^r(1-z)^{m-1-r}/((1-z)^m - y^m)$, which completes our first proof of Theorem 1.

Remark 2. We can get an even more refined generating function by using a different variable for every transition in the DFA. Specifically, for $0 \leq i < m$, let the power of z_i (resp. y_i) count the number of occurrences of 0's (resp. 1's) in a word such that the number of 1's strictly preceding this occurrence is congruent to i mod m. The analysis given above proves that

$$\begin{aligned} &G_{m,r}(y_0, \ldots, y_{m-1}, z_0, \ldots, z_{m-1}) \\ &\qquad = \frac{y_0 y_1 \cdots y_{r-1}(1-z_{r+1})(1-z_{r+2}) \cdots (1-z_{m-1})}{(1-z_0)(1-z_1)\cdots(1-z_{m-1}) - y_0 y_1 y_2 \cdots y_{m-1}}. \end{aligned} \tag{10}$$

Theorem 1 is obtained by specializing all z_i to z and all y_i to y.

2.2. Bijective Proof of Theorem 1

Clearing denominators in (8) and rearranging terms, we can rewrite Theorem 1 in the equivalent form

$$(1-z)^m G_{m,r}(y,z) = y^r(1-z)^{m-1-r} + y^m G_{m,r}. \tag{11}$$

We now give a bijective proof of this identity. The proof consists of three steps: first, model the left side of (11) by a set X of signed, weighted objects; second, introduce a sign-reversing, weight-preserving involution $I : X \to X$ that cancels most of the objects; third, show that the right side of (11) is the generating function for the remaining uncancelled objects.

For the first step, let X consist of all pairs (v,w), where $v = v_0 v_1 \cdots v_{m-1} \in \{e,0\}^m$ is a sequence of e's and 0's of length m, and w is a word in $W_{*,m,r}$. Here, e stands for "empty." Define the *sign* of (v,w) to be $(-1)^{N_0(v)}$, so the object (v,w) is negative iff v contains an odd number of 0's. Define the *weight* of (v,w) to be $y^{N_1(w)} z^{N_0(v)+N_0(w)}$. For example, when $m = 5$ and $r = 2$, the object $(ee0e0, 11101001011)$ in X has sign $(-1)^2 = +1$ and weight $y^7 z^6$. In general, we can build $v \in \{e,0\}^m$ by choosing an e (with signed weight 1) or a 0 (with signed weight $-z$) in each of m positions. Recall $G_{m,r}$ is the generating function for the set $W_{*,m,r}$. It follows that the generating function for X, namely $\sum_{(v,w)\in X} \operatorname{sgn}(v,w)\operatorname{wt}(v,w)$, is indeed given by $(1-z)^m G_{m,r}(y,z)$.

For the second step, we define the involution $I : X \to X$ as follows. Given $(v, w) \in X$, write $w \in W_{*,m,r}$ in the "factored" form

$$w = 0^{a_0}10^{a_1}10^{a_2}1 \cdots 10^{a_r}10^{a_{r+1}} \cdots 0^{a_{m-1}}1w',$$

where: 0^{a_i} denotes a string of a_i copies of 0 with $a_i \geq 0$; w' is the part of w following the m'th 1 in w (possibly empty); and the part of the factorization starting $10^{a_{r+1}}$ is not present if w only has r copies of 1. In the latter case, $a_{r+1}, \ldots, a_{m-1}, w'$ are undefined. To compute $(v', w') = I(v, w)$, look for the least $i < m$ such that a_i is defined and ($v_i = 0$ or $a_i > 0$). If no such i exists, define $I(v, w) = (v, w)$. If i exists and $v_i = 0$, replace v_i by e and add one more 0 to the string 0^{a_i} in w. If i exists and $v_i = e$, replace v_i by 0 and remove one 0 from the string 0^{a_i} in v. It is immediate that I preserves weights and (for non-fixed points) reverses signs. Furthermore, doing I twice in succession restores the original object, so I is an involution. For our example object above, $a_0 = a_1 = a_2 = 0$, $a_3 = 1$, $a_4 = 2$, $w' = 011$, so $i = 2$. I sends this object to $(eeee0, 110101001011)$, which has signed weight $-y^7z^6$. As another example, $I(e000e, 00010010) = (0000e, 0010010)$.

For the third step, we examine the fixed points of I. We ask: for which objects (v, w) does i not exist? This can happen in two ways. First, if w has only r copies of 1 (so that $a_{r+1}, \ldots, a_{m-1}, w'$ are undefined) and $v_0 = \cdots = v_r = e$ and $a_0 = \cdots = a_r = 0$, then i does not exist. (An example when $m = 5$ and $r = 2$ is $(eee0e, 11)$.) In this case w must be 1^r, and the last $m-1-r$ symbols of v could each be e or 0. So the generating function for these fixed points is $y^r(1-z)^{m-1-r}$. Second, if w has at least m copies of 1 and $v_0 = \cdots = v_{m-1} = e$ and $a_0 = \cdots = a_{m-1} = 0$, then i does not exist. (An example when $m = 5$ and $r = 2$ is $(eeeee, 1111100101)$.) In this case, w must begin with the string 1^m, and deleting this string from w gives an arbitrary word $w' \in W_{*,m,r}$ counted by $G_{m,r}$. Thus, the generating function for these fixed points is $y^mG_{m,r}$. Combining the expressions for the two types of fixed points gives the right side of (11). This completes the bijective proof of Theorem 1. The same argument proves the refined version (10), as is readily checked.

2.3. Extensions of Theorem 1

Theorem 1 can be extended to give the generating function for certain sums of multinomial coefficients. For example, we can prove

$$\sum_{\substack{a,b,c,d \geq 0: \\ (a+b) \bmod m = r}} \binom{a+b+c+d}{a, b, c, d} x_1^a x_2^b x_3^c x_4^d = \frac{(x_1 + x_2)^r(1 - x_3 - x_4)^{m-1-r}}{(1 - x_3 - x_4)^m - (x_1 + x_2)^m}$$

for $0 \leq r < m$. The left side is a four-variable generating function for words in $\{1, 2, 3, 4\}^*$, where the power of x_i counts the number of i's in the word, and the total number of 1's and 2's in the word must be congruent to r mod m. To prove this, modify $\mathrm{DFA}_{m,r}$ by replacing each loop transition labeled 0 with two loops labeled 3 and 4, and adding new transitions labeled 2 from state i to state $i+1 \bmod m$. The bijective proof readily extends to this situation. We can also refine the generating function by using a different variable for each transition.

Next we consider extensions of Theorem 1 to q-binomial coefficients. For any binary word w, the *inversion count* $\mathrm{inv}(w)$ is the number of pairs $i<j$ with $w_i=1$ and $w_j=0$. The *q-binomial coefficient* $\left[{n \atop k}\right]_q$ can be defined combinatorially as the sum of $q^{\mathrm{inv}(w)}$ over all words w consisting of k copies of 1 and $n-k$ copies of 0, where q is a formal variable. The bijective proof of the refined generating function (10) extends to this setting, leading to the formula:

$$\begin{aligned}(1-z_0)(1-qz_1)\cdots(1-q^{m-1}z_{m-1})&G_{m,r}(y_0,\ldots,y_{m-1},z_0,\ldots,z_{m-1};q)\\ &= y_0\cdots y_{r-1}(1-q^{r+1}z_{r+1})\cdots(1-q^{m-1}z_{m-1})\\ &\quad + y_0\cdots y_{m-1}G_{m,r}(y_0,\ldots,y_{m-1},q^m z_0,\ldots,q^m z_{m-1};q). \qquad (12)\end{aligned}$$

Extra powers of q have been added to ensure that the involution I still preserves the q-weight of objects $(v,w)\in X$. To see how these powers arise, consider the case where I changes v_i from 0 to e and increases a_i by 1. This action puts one new 0 after the i'th 1 in w, which increases the inversion count of w by exactly i. To balance this, we must weight a 0 in position i of v by q^i, explaining why each factor $1-z_i$ becomes $1-q^i z_i$ in (12). Next consider fixed points of I of the form (e^m,w) where $w=1^m w'$. If w' has c zeroes, then $\mathrm{inv}(w)=mc+\mathrm{inv}(w')$ due to the m initial 1's in w. To account for the extra power q^{mc}, we must replace each variable z_i by $q^m z_i$ in the generating function for w' appearing in the third line of (12).

We can obtain a more concise formula by specializing the formal variable q to be $\omega=e^{2\pi i/m}$ (or any other m'th root of unity), which eliminates the extra powers q^m. Specializing every y_i and z_i to x, we obtain the generating function

$$\sum_{n\geq 0}\sum_{k\geq 0}\begin{bmatrix} n \\ km+r\end{bmatrix}_\omega x^n = \frac{x^r(1-\omega^{r+1}x)\cdots(1-\omega^{m-1}x)}{(1-x)(1-\omega x)\cdots(1-\omega^{m-1}x)-x^m}$$

for $0\leq r<m$.

3. The Minimum Sums $\mu(n,m)$

This section studies the *minimum sums*

$$\mu(n,m)=\min_{r:\ 0\leq r<m}\sum_{k\geq 0}\binom{n}{km+r}. \qquad (13)$$

Our main goal is a bijective proof of the formula $\mu(n,m)=(2^n-\ell(n,m))/m$, where $\ell(n,m)$ counts strings of left and right parentheses of length n such that every prefix has at most $m-2$ unmatched parentheses within that prefix. But first, we describe a variation of Pascal's Triangle that lets us compute particular sums $S(n,m,r)$ recursively. This allows us to identify which r's attain the minimum sum in (13).

3.1. Pascal's Triangle on a Cylinder

For fixed $m > 0$, we can compute all the sums $S(n,m,r)$ for $n \geq 0$ and $0 \leq r < m$ by the following construction. Form an initially empty array with rows indexed by $n = 0, 1, 2, \ldots$, and columns indexed by $r = 0, 1, \ldots, m-1$. The entry in row n and column r of this array will be $S(n,m,r)$. Start by filling in row 0 with 1 followed by $m-1$ zeroes. Having already filled in row $n-1$, fill in row n via the rule $S(n,m,r) = S(n-1,m,r) + S(n-1,m,r-1)$, where $r-1$ is reduced mod m. This rule follows immediately from the definition of $S(n,m,r)$ and Pascal's recursion $\binom{n}{a} = \binom{n-1}{a} + \binom{n-1}{a-1}$ for binomial coefficients. In visual terms, we get the entry in column 0 of the new row by adding the entries in the leftmost column and the rightmost column of the previous row. In every other column of the new row, we get the new entry by adding the entries directly above it and northwest of it. The beginning of the array for $m = 5$ is shown below, where the minimum sum(s) in each row $n \geq 4$ are underlined:

n	$r=0$	$r=1$	$r=2$	$r=3$	$r=4$
0	1	0	0	0	0
1	1	1	0	0	0
2	1	2	1	0	0
3	1	3	3	1	0
4	$\underline{1}$	4	6	4	$\underline{1}$
5	$\underline{2}$	5	10	10	5
6	$\underline{7}$	$\underline{7}$	15	20	15
7	22	$\underline{14}$	22	35	35
8	57	$\underline{36}$	$\underline{36}$	57	70
9	127	93	$\underline{72}$	93	127
10	254	220	$\underline{165}$	$\underline{165}$	220
11	474	474	385	$\underline{330}$	385
12	859	948	859	$\underline{715}$	$\underline{715}$

We now ask: which r's attain the minimum in $\mu(n,m) = \min_r S(n,m,r)$? The pattern in the table above suggests that these r's increase (mod m) at rate 1/2 compared to n. Starting in the appropriate column, the entries in each row also appear to increase from the minimum sum to the maximum sum and then decrease through the same values. The next theorem gives a precise statement of these observations.

Theorem 3. *Let* $r_1(n,m) = \lceil (n-m)/2 \rceil \bmod m$ *and* $r_2(n,m) = \lfloor (n-m)/2 \rfloor \bmod m$. *For* $n \geq m-1$, $\mu(n,m) = S(n,m,r)$ *if and only if* $r = r_1(n,m)$ *or* $r = r_2(n,m)$.

Moreover, the m entries in row n of the Pascal array (starting in column $r_1(n,m)$ and wrapping around at the end) have the form shown in the table below, where $a_1 < a_2 < \ldots < a_{h+1}$.

Parity of m	Parity of n	Row n starting in column $r_1(n,m)$
$m = 2h$ is even	n is odd	$a_1, a_2, \ldots, a_h, a_h, \ldots, a_2, a_1$
$m = 2h$ is even	n is even	$a_1, a_2, \ldots, a_h, a_{h+1}, a_h, \ldots, a_2$
$m = 2h+1$ is odd	n is even	$a_1, a_2, \ldots, a_h, a_{h+1}, a_h, \ldots, a_2, a_1$
$m = 2h+1$ is odd	n is odd	$a_1, a_2, \ldots, a_h, a_{h+1}, a_{h+1}, a_h, \ldots, a_2$

Proof. For fixed m, all claims in the theorem are routinely proved simultaneously by induction on $n \geq m-1$. The base case $n = m-1$ follows from known unimodality and symmetry properties of binomial coefficients, since row $m-1$ of the m-column array agrees with row $m-1$ of the ordinary Pascal's Triangle. To avoid cumbersome notation, we illustrate one case of the induction step when $m = 7$ (although the argument is completely general). Suppose the theorem is already known for some odd $n \geq m-1$; we prove the theorem for the even row $n+1$. Starting in column $r_1(n,m)$, row n of the 7-column array looks like a, b, c, d, d, c, b where $a < b < c < d$ (by induction hypothesis). Applying the recursion, we see that row $n+1$ (still starting in column $r_1(n,m)$) looks like $a', a', b', c', d', c', b'$ where $a' = a+b$, $b' = b+c$, $c' = c+d$, $d' = 2d$, and $a' < b' < c' < d'$. Now $r_1(n+1,m) = r_1(n,m)+1$ in this case, so row $n+1$ starting in column $r_1(n+1,m)$ looks like $a', b', c', d', c', b', a'$. This has the form in row 3 of the table, and the minimum sum in this row (namely a') does occur in columns $r_1(n+1,m)$ and $r_2(n+1,m) = r_1(n,m)$, as needed. The other cases are checked in the same way. □

3.2. The Optimal Bijection

Fix m and $n \geq m-1$. We now construct the sets $D, A_0, A_1, \ldots, A_{m-1}, L$ and the bijection $f : D \to D$ with the properties stated below (6). Here, $D = \{0,1\}^n \setminus L = \bigcup_{r=0}^{m-1} A_r$. Throughout this discussion, we regard binary strings $w \in \{0,1\}^n$ as strings of parentheses by interpreting each 0 as a *right* parenthesis and each 1 as a *left* parenthesis. For any binary word w, let $U(w)$ be the maximum number of unmatched parentheses in any prefix of w. Here and below, when counting how many parentheses in a prefix are unmatched, we look only at symbols within that prefix. For instance, the prefix `((()` of the word `((()))` has two unmatched parentheses. The U-value of this prefix and the overall word is 3. Define the *leftover set*

$$L = L(n,m) = \{w \in \{0,1\}^n : U(w) < m-1\}.$$

For $0 \leq r < m$, define $A_r = W_{n,m,r} \setminus L$. So, a binary word $w \in \{0,1\}^n$ belongs to A_r if and only if $N_1(w) \equiv r \pmod{m}$ and there exists a prefix of w with at least $m-1$ unmatched parentheses. Since adding one symbol to a prefix increases the number of unmatched parentheses by at most 1, we see that some prefix of $w \in A_r$ has exactly $m-1$ unmatched parentheses.

Clearly, $\{0,1\}^n$ is the disjoint union of $A_0, \ldots, A_{m-1}, L$. Given $w \in A_r$, define $f(w)$ as follows. Find the shortest prefix of w with exactly $m-1$ unmatched parentheses; say this prefix has length p. The subword of unmatched parentheses in this prefix must have the form $)^a\,(^{m-1-a}$. Change this subword to $)^{a-1}\,(^{m-1-(a-1)}$ interpreting exponents mod m. Then $N_1(f(w)) \equiv N_1(w) + 1 \pmod{m}$, as needed. Now suppose we apply f to $f(w)$. We claim that the same shortest prefix length p will be found, and the subword of unmatched parentheses within this prefix will be in the same positions in $f(w)$ as in w. We verify this claim following the example below. Using the claim repeatedly, we see that iterating f causes the subword in question to cycle through the m words of the form $)^b\,(^{m-1-b}$, for $0 \le b < m$. In particular, $f^m(w) = w$, as needed. Thus, $f : D \to D$ is a bijection satisfying (4) for all $w \in D$.

Example 4. Let $n = 17$ and $m = 5$. Given the word $w = 01101110100111010$, first rewrite w as $)(()((()())((()()$. Next, scan prefixes and look for unmatched parentheses (which are underlined below):

prefix length	prefix	unmatched count
1	$\underline{)}$	1
2	$\underline{)(}$	2
3	$\underline{)((}$	3
4	$\underline{)(}()$	2
5	$\underline{)(}()\underline{(}$	3
6	$\underline{)(}()\underline{((}$	4

The substring of unmatched parentheses in the prefix of length six looks like $\underline{)(((}$. Repeatedly applying the map f changes this substring as follows:

$$\underline{)(((} \xmapsto{f} \underline{((((} \xmapsto{f} \underline{))))} \xmapsto{f} \underline{)))(} \xmapsto{f} \underline{))((} \xmapsto{f} \underline{)(((}.$$

Therefore

$$w = \underline{)(}()\underline{((}()())((()() \xmapsto{f} \underline{((}()\underline{((}()())((()() \xmapsto{f} \underline{))}()\underline{))}()())((()()$$

$$\xmapsto{f} \underline{))}()\underline{)(}()())((()() \xmapsto{f} \underline{))}()\underline{((}()())((()() \xmapsto{f} \underline{)(}()\underline{((}()())((()() = w.$$

Now we prove the claim. For concreteness, we consider the case $m = 8$ and $a = 3$, although the argument is perfectly general. The critical prefix for w can be written in the form

$$w_1 w_2 \cdots w_p \quad = \quad w^{(1)}\,\underline{)}\,w^{(2)}\,\underline{)}\,w^{(3)}\,\underline{)}\,w^{(4)}\,\underline{(}\,w^{(5)}\,\underline{(}\,w^{(6)}\,\underline{(}\,w^{(7)}\,\underline{(}, \qquad (14)$$

where each $w^{(i)}$ is a binary word, and the displayed underlined parentheses are *all* the unmatched parentheses in this prefix. Clearly, the final symbol in the prefix must be an unmatched parenthesis, by minimality of p. Define $M_{\leq i}$ to be the set of binary words w such that: $U(w) \leq i$, the full word w is *balanced* (i.e., has no unmatched parentheses), and no prefix of w has an unmatched right parenthesis. In particular, $M_{\leq 0}$ consists of the empty word alone.

The key observation about (14) is that we must have $w^{(i)} \in M_{\leq 7-i}$ for $1 \leq i \leq 7$. Consider $w^{(1)}$ first. An unmatched right parenthesis in $w^{(1)}$ (or any of its prefixes) would remain unmatched in any longer prefix of w, contradicting the assumption that *all* unmatched parentheses in $w_1 \cdots w_p$ are displayed separately in (14). An unmatched left parenthesis in $w^{(1)}$ would match with the first displayed right parenthesis in (14), contradicting the assumption that that right parenthesis is unmatched. Thus, $w^{(1)}$ must be balanced. Also, $U(w^{(1)}) \leq 6$ by minimality of p, since otherwise some prefix of $w^{(1)}$ would be the shortest prefix of w with 7 unbalanced parentheses. Thus, $w^{(1)} \in M_{\leq 6}$. The same argument shows that $w^{(2)}$ must be balanced, and no prefix of $w^{(2)}$ has any unmatched right parentheses. Moreover, since there is already one unmatched right parenthesis prior to $w^{(2)}$ in w (which can never be matched with any later symbol), we must have $U(w^{(2)}) \leq 5$ by minimality of p. Thus, $w^{(2)} \in M_{\leq 5}$. Similarly, $w^{(3)} \in M_{\leq 4}$.

Next consider $w^{(4)}$. As before, there cannot be any unmatched right parenthesis in $w^{(4)}$ or any of its prefixes, or this parenthesis would have been displayed separately in (14). Every prefix of $w^{(4)}$ has at most 3 unmatched left parentheses, by minimality of p. Finally, the full word $w^{(4)}$ cannot have any unmatched left parenthesis. For, such a left parenthesis must later be matched with a right parenthesis in $w_1 \cdots w_p$ coming from $w^{(5)}$, $w^{(6)}$, or $w^{(7)}$. But this cannot happen, since such a right parenthesis would have instead matched with the displayed (unmatched) left parenthesis following $w^{(4)}$. Thus, $w^{(4)} \in M_{\leq 3}$. When we reach $w^{(5)}$, we see that $w^{(5)}$ and its prefixes cannot have any unmatched right parentheses, since one of these would match with the displayed (unmatched) left parenthesis immediately preceding $w^{(5)}$. The same argument used for $w^{(4)}$ shows that $w^{(5)}$ is balanced and $U(w^{(5)}) \leq 2$, so that $w^{(5)} \in M_{\leq 2}$. We similarly show that $w^{(6)} \in M_{\leq 1}$ and $w^{(7)} \in M_{\leq 0}$ (in particular, $w^{(7)}$ must be empty).

When we apply f to w, (14) changes to the prefix

$$w'_1 w'_2 \cdots w'_p \quad = \quad w^{(1)} \,\underline{)}\, w^{(2)} \,\underline{)}\, w^{(3)} \,\underline{(}\, w^{(4)} \,\underline{(}\, w^{(5)} \,\underline{(}\, w^{(6)} \,\underline{(}\, w^{(7)} \,\underline{(} \tag{15}$$

of $f(w)$. Now that we know each $w^{(i)}$ is balanced with $U(w^{(i)}) \leq 7 - i$, we can check by inspection that every proper prefix of (15) has fewer than 7 unmatched parentheses. Moreover, the full prefix shown has exactly 7 unmatched parentheses, which are exactly the ones displayed separately and underlined. Thus, when we apply f to $f(w)$, the map acts on the same set of 7 displayed positions. This completes the proof of the claim. It is easy to check that our argument works for all

m and a, including the extreme cases $a = 0$ (where the displayed prefix (14) has no unmatched right parentheses) and $a = m - 1$ (where this prefix has no unmatched left parentheses).

3.3. Proof of Optimality

We must still prove that our bijection f satisfies the *optimality property* $|A_r| = \mu(n, m)$ for $0 \le r < m$. Recall that each A_r is a subset of $W_{n,m,r}$ and $N_1(f(w)) \equiv N_1(w) + 1 \bmod m$ for all $w \in A_r$. The restriction of f to A_r is a bijection from A_r to $A_{r+1 \bmod m}$ for each r, so the sets $A_0, A_1, \ldots, A_{m-1}$ all have the same size. It suffices to show that there exists an r (which must, in fact, be $r_1(n, m)$ or $r_2(n, m)$ from Theorem 3) such that A_r is the whole set $W_{n,m,r}$.

Our proof will use induction on n, holding m fixed. For a binary word $w \in \{0, 1\}^n$ of any length n, let us say that w is *good mod m* if and only if w is in the domain $\{0, 1\}^n \setminus L(n, m)$ of the bijection f constructed above. This means that some prefix of w has exactly $m - 1$ unmatched parentheses.

Lemma 1. *Suppose w is good mod m and w' is obtained from w by adding two consecutive symbols* 10 *anywhere in w. Then w' is good mod m.*

Proof. Fix a word w that is good mod m. Consider the shortest prefix $w_1 w_2 \cdots w_p$ of w with exactly $m - 1$ unmatched parentheses, as in (14). We get w' from w by adding consecutive symbols () at some position. If this position is after w_p, then $w_1 \cdots w_p$ is also a prefix of w', so that w' is good mod m in this case. Otherwise, we obtain w' from w by adding a consecutive pair of matched parentheses to one of the words $w^{(i)}$ shown in (14). The new $w^{(i)}$ is still balanced. If the new $w^{(i)}$ is still in $M_{\le m-1-i}$, then the prefix of w' ending at w_p (now of length $p + 2$) still has exactly $m - 1$ unmatched parentheses. It is also possible that the new $w^{(i)}$ now has $U(w^{(i)}) = m - i$. But in this event, some prefix of w' ending before w_p has exactly $m - 1$ unmatched parentheses. Thus, in all cases, w' is still good mod m. □

The converse of the lemma is false: for example, taking $w = 0010$ and $w' = 001100$, w' is good mod 5 but w is not.

Theorem 5. *For all $n \ge m - 1$, every word in $W_{n,m,r_1(n,m)}$ and $W_{n,m,r_2(n,m)}$ is good mod m.*

Proof. We fix m and use induction on $n \ge m - 1$. For the base case $n = m - 1$, we have $r_1 = 0$ and $r_2 = m - 1$, so we must show that every word in $W_{m-1,m,0}$ and $W_{m-1,m,m-1}$ is good mod m. The only word in $W_{m-1,m,0}$ is 0^{m-1} (a string of $m - 1$ right parentheses), and the only word in $W_{m-1,m,m-1}$ is 1^{m-1} (a string of $m - 1$ left parentheses). These words are certainly good mod m. For the base case $n = m$, we have $r_1 = r_2 = 0$. Here, $W_{m,m,0} = \{0^m, 1^m\}$, and these two words are both good mod m.

Now fix $n \geq m+1$. By induction, we may assume that all words in $W_{n-2,m,r}$ and $W_{n-2,m,r'}$ are good mod m, where $r = r_1(n-2,m)$ and $r' = r_2(n-2,m)$. Since $r_1(n,m) = r+1$ and $r_2(n,m) = r'+1$ (taking all additions mod m), we must prove that all words in $W_{n,m,r+1}$ and $W_{n,m,r'+1}$ are good. Let w' be an arbitrary word in $W_{n,m,r+1}$. If w' has no consecutive substring 10, then w' must have the form $0^a 1^{n-a}$ (i.e., a right parentheses followed by $n-a$ left parentheses), and this word is certainly good mod m. Otherwise, let w be the word obtained from w' by deleting the first occurrence of the substring 10. Then $w \in W_{n-2,m,r}$, hence w is good mod m by induction hypothesis, hence w' is good mod m by Lemma 1. The same argument works for words $w' \in W_{n,m,r'+1}$. □

This theorem proves that all the bijections f are optimal, completing our bijective proof of (7).

4. Analysis of $L(n,m)$

We have characterized the "error term" $\ell(n,m)$ in (7) as counting the set $L(n,m)$ of words w in $\{0,1\}^n$ such that every prefix of w has at most $m-2$ unmatched parentheses. It is natural to ask for other descriptions of $\ell(n,m)$ such as explicit formulas or generating functions. On one hand, by comparing (7) and (2) and using Theorem 3, we have

$$\ell(n,m) = -2^{n+1} \sum_{k=1}^{\lfloor (m-1)/2 \rfloor} \cos^n \frac{\pi k}{m} \cos \frac{\pi k(n-2r')}{m}$$

when $r' = r_1(n,m)$ or $r' = r_2(n,m)$. On the other hand, as noted in the Introduction, for small values of m these strange trigonometric expressions simplify to integer sequences with predictable structure.

In this section, we use more DFAs to derive recursions for the generating functions $\sum_{n\geq 0} \ell(n,m)x^n$ and their bivariate analogues. When $m=5$, this leads to a surprising combinatorial interpretation for odd-indexed Lucas numbers.

4.1. DFA for the Language $M_{\leq k}$

Recall that $M_{\leq k}$ is the set (language) of balanced binary words w such that every prefix of w has no unmatched right parentheses and at most k unmatched left parentheses. Define the generating functions

$$B_k(y,z) = \sum_{w \in M_{\leq k}} y^{N_1(w)} z^{N_0(w)}.$$

Each $M_{\leq k}$ is a regular language recognized by a DFA denoted DFA$_{\leq k}$. This DFA has state space $\{0,1,2,\ldots,k,\Omega\}$, where 0 is the start state and Ω is a "death state."

For $0 \leq i < k$, there is a transition labeled 1 (left parenthesis) from state i to state $i+1$. For $0 < i \leq k$, there is a transition labeled 0 (right parenthesis) from state i to state $i-1$. We can also transition from state k to state Ω via symbol 1, from state 0 to state Ω via symbol 0, or from state Ω to itself via symbols 0 and 1. State 0 is the only accepting state. The automaton DFA$_{\leq 4}$ is drawn in Figure 2; the death state is not shown.

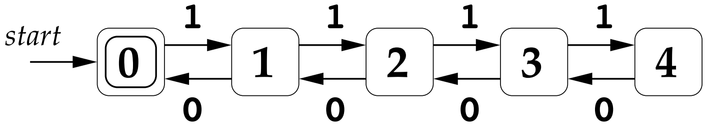

Figure 2: DFA recognizing $M_{\leq 4}$.

It is routine to check that DFA$_{\leq k}$ recognizes the language $M_{\leq k}$. The key point is that this DFA reaches state $i \neq \omega$ upon reading a given prefix of input w only if that prefix has i unmatched left parentheses. We enter state ω and never leave if and only if some prefix of the input has an unmatched right parenthesis or too many unmatched left parentheses. Finally, since the only accepting state is state 0, we only accept inputs where the full word is balanced.

We could find a formula for $B_k(y,z)$ by inverting a matrix, as we did in §2.1. However, a simpler approach is based on the observation that there is a "copy" of DFA$_{\leq k-1}$ inside DFA$_{\leq k}$ to the right of state 0. We can classify all words w in $M_{\leq k}$ based on how many times the transition from state 0 to state 1 is taken on input w. If this transition is taken $j \geq 0$ times, then w can be uniquely factorized

$$w = 1w^{(1)}0\ 1w^{(2)}0\ 1w^{(3)}0\ \cdots\ 1w^{(j)}0,$$

where $w^{(1)}, \ldots, w^{(j)}$ are strings in $M_{\leq k-1}$. Passing to generating functions, it follows that

$$B_k(y,z) = \sum_{j=0}^{\infty} (yB_{k-1}(y,z)z)^j \qquad \text{for } k > 0.$$

Summing this formal geometric series, we get the recursion and initial condition

$$B_k(y,z) = \frac{1}{1 - yzB_{k-1}(y,z)} \qquad \text{for } k > 0; \qquad B_0(y,z) = 1. \tag{16}$$

We can interpret this as a formal continued fraction. It is routine to check (by induction on k) that we can write $B_k(y,z) = p_{k-1}(y,z)/p_k(y,z)$ for all $k > 0$, where $(p_k(y,z) : k \geq 0)$ is the sequence of polynomials defined recursively by

$$p_0 = 1,\ p_1 = 1 - yz,\ p_k = p_{k-1} - yzp_{k-2} \qquad \text{for } k \geq 2.$$

Another induction proof shows that

$$p_k(y,z) = \sum_{i\geq 0} \binom{k+1-i}{i} (-yz)^i \quad \text{for } k \geq 0. \tag{17}$$

We need the following variant of the languages $M_{\leq k}$. Modify DFA$_{\leq k}$ by making every state an accepting state except for the death state Ω. The new DFA recognizes the language $M^p_{\leq k}$ consisting of all prefixes of words in $M_{\leq k}$. Equivalently, a binary word w is in $M^p_{\leq k}$ iff every prefix of w has no unmatched right parentheses and at most k unmatched left parentheses, but w itself need not be balanced. Let $A_k(y,z) = \sum_{w\in M^p_{\leq k}} y^{N_1(w)} z^{N_0(w)}$ be the generating function for $M^p_{\leq k}$. Each word $w \in M^p_{\leq k}$ either is in $M_{\leq k}$ or can be factored uniquely as $w = u1v$, where $u \in M_{\leq k}$ and $v \in M^p_{\leq k-1}$. Here, the displayed 1 is the last symbol in w causing DFA$_{\leq k}$ to transition from state 0 to state 1. We deduce the recursion and initial condition

$$A_k = B_k(1 + yA_{k-1}) \quad \text{for } k > 0; \qquad A_0 = 1. \tag{18}$$

By solving this equation for B_k, inserting the resulting expressions into (16), and simplifying, we are led to the recursion

$$A_k = \frac{(1+yA_{k-1})(1+yA_{k-2})}{1+yA_{k-2}-yzA_{k-1}} \quad \text{for } k \geq 2; \qquad A_0 = 1, \quad A_1 = \frac{1+y}{1-yz}.$$

Using (18) and $B_k = p_{k-1}/p_k$, one sees by induction on k that each A_k is a rational function with denominator $p_k(y,z)$.

Remark 6. By looking at how input words move through the states of DFA$_{\leq k}$, we can regard the languages $M_{\leq k}$ and $M^p_{\leq k}$ as encoding certain random walks. Interpret 1 (resp. 0) as a unit step right (resp. left) on a number line.[2] Words $w \in M_{\leq k}$ encode random walks starting and ending at $x = 0$ that never visit $x = -1$ or $x = k+1$. For $M^p_{\leq k}$, the requirement of ending at $x = 0$ is dropped.

4.2. DFA for the Language $L_{\leq k}$

Let $L_{\leq k}$ be the set (language) of binary words w such that every prefix of w has at most k unmatched parentheses. Then $L(n,m)$ is the set of words of length n in $L_{\leq m-2}$ and $\ell(n,m) = |L(n,m)|$. This section describes a DFA recognizing $L_{\leq k}$ and derives a recursion for the generating function

$$C_k(y,z) = \sum_{w\in L_{\leq k}} y^{N_1(w)} z^{N_0(w)}.$$

The DFA recognizing $L_{\leq k}$ is obtained by "gluing together" vertical copies of DFA$_{\leq i}$ for $i = k, k-1, \ldots, 2, 1, 0$, as shown in Figure 3 for the case $k = 3$. For each

[2] So a right step corresponds to a left parenthesis, and a left step corresponds to a right parenthesis.

(i,j) with $i+j \leq k$, there is an accepting state labeled (i,j). State $(0,0)$ is the start state. Transitions are as indicated in the figure. Transitions not shown in the figure go from states (i,j) with $i+j=k$ to a non-accepting death state Ω.

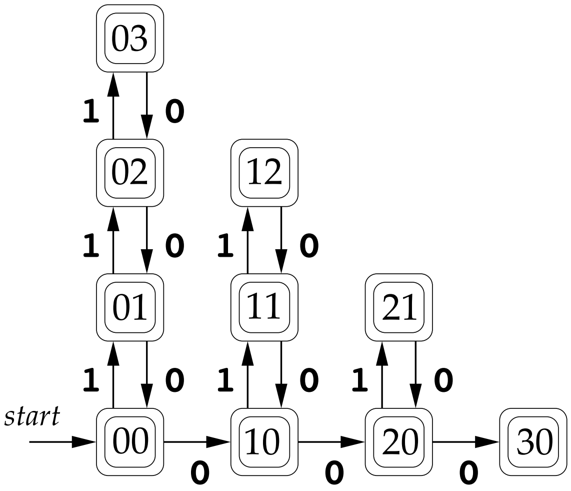

Figure 3: DFA recognizing $L_{\leq 3}$.

It is routine to check that we reach state (i,j) after reading some prefix v of an input w if and only if v has i unmatched right parentheses and j unmatched left parentheses, and all prefixes of v have at most k unmatched parentheses (compare to our analysis of (14)). Hence, this DFA does recognize the language $L_{\leq k}$.

Theorem 7. *The generating functions $C_k(y,z)$ for the languages $L_{\leq k}$ satisfy the recursion and initial condition*

$$C_k = A_k + zB_kC_{k-1} = B_k(1+yA_{k-1}+zC_{k-1}) \quad \textit{for } k>0; \quad C_0 = 1;$$

where A_k and B_k are characterized by (18) *and* (16)*. Each C_k is a rational function with denominator $p_k(y,z)$ given by* (17)*.*

Proof. For $k>0$, a word $w \in L_{\leq k}$ either belongs to $M^p_{\leq k}$ or can be uniquely factorized $w = u0v$ where $u \in M_{\leq k}$ and $v \in L_{\leq k-1}$. The displayed 0 (weighted by z) is the symbol causing the DFA to transition from state $(0,0)$ to state $(1,0)$. Passing to generating functions, we immediately obtain the recursion $C_k = A_k + zB_kC_{k-1}$. The second formula for C_k follows from this one using (18). The final statement of the theorem follows from the recursion by induction on k, since $B_k = p_{k-1}/p_k$ and A_k has denominator p_k. □

One consequence of our DFA analysis is that for all n and m,

$$\text{if } n < m-2 \text{ or } n-m \text{ is odd, then } \ell(n+1,m) = 2\ell(n,m). \tag{19}$$

To prove this, consider an arbitrary word $w \in L(n,m) \subseteq L_{\leq m-2}$. Because of the assumption on m and n, this word must be accepted by a state (i,j) in the DFA with $i+j \neq m-2$. Adding either of the two symbols 0 or 1 to the end of w leads from this state to another accepting state of the DFA. So we have a bijection from $L(n,m) \times \{0,1\}$ to $L(n+1,m)$, as needed.

Remark 8. Words in $L_{\leq k}$ encode random walks satisfying these restrictions: the walk starts at $x=0$; the walk never goes left of $x=-k$; and at all times, the walk is never more than k units right of the least x-coordinate visited so far.

4.3. Connection to Lucas Numbers

Using the recursions derived earlier, we compute

$$\begin{aligned}
&B_0 = 1, \quad B_1 = \tfrac{1}{1-yz}, \quad B_2 = \tfrac{1-yz}{1-2yz}, \quad B_3 = \tfrac{1-2yz}{1-3yz+y^2z^2}, \\
&A_0 = 1, \quad A_1 = \tfrac{1+y}{1-yz}, \quad A_2 = \tfrac{1+y+y^2-yz}{1-2yz}, \quad A_3 = \tfrac{1+y+y^2+y^3-2yz-y^2z}{1-3yz+y^2z^2}, \\
&C_0 = 1, \quad C_1 = \tfrac{1+y+z}{1-yz}, \quad C_2 = \tfrac{1+y+y^2+z+z^2}{1-2yz}, \quad C_3 = \tfrac{1+y+y^2+y^3+z-yz+z^2+z^3}{1-3yz+y^2z^2}.
\end{aligned}$$

Using these generating functions (setting $y=z=x$), it is routine to confirm the values for $\ell(n,2)$, $\ell(n,3)$, and $\ell(n,4)$ stated in the Introduction. We now use C_3 to explain the formula for $\ell(n,5)$ in terms of Lucas numbers.

Theorem 9. *For all even $n \geq 0$, $\ell(n,5) = \mathrm{Luc}_{n+1}$. For all odd $n > 0$, $\ell(n,5) = 2\,\mathrm{Luc}_n$.*

Proof. The statement for odd n follows from the statement for even n and (19), so we focus on the case of even n. Start with $F(x) = C_3(x,x) = \sum_{n\geq 0} \ell(n,5)x^n = \frac{1+2x+x^2+2x^3}{1-3x^2+x^4}$. To isolate the even powers of x, we compute

$$\frac{F(x)+F(-x)}{2} = \sum_{n \text{ even}} \ell(n,5)x^n = \frac{1+x^2}{1-3x^2+x^4}.$$

On the other hand, the generating function for Lucas numbers is

$$G(x) = \sum_{n\geq 0} \mathrm{Luc}_n\, x^n = \frac{2-x}{1-x-x^2}.$$

Isolating the odd powers of x gives

$$\frac{G(x)-G(-x)}{2} = \sum_{n \text{ odd}} \mathrm{Luc}_n\, x^n = x \sum_{n \text{ even}} \mathrm{Luc}_{n+1}\, x^n = \frac{x+x^3}{1-3x^2+x^4}.$$

Dividing by x, we see that the two generating functions agree. □

To the authors' knowledge, the combinatorial interpretation for odd-indexed Lucas numbers in Theorem 9 has not appeared before. It would be interesting to prove this theorem by finding a bijection between the set of words $L(n, 5)$ and one of the standard collections of objects counted by the Lucas numbers (see, for instance, [7]).

References

[1] Arthur Benjamin, Bob Chen, and Kimberly Kindred, Sums of evenly spaced binomial coefficients, *Math. Mag.* **83** no. 5 (2010), 370–373.

[2] Arthur Benjamin and Jacob Scott, Third and fourth binomial coefficients, *Fibonacci Quart.* **49** no. 2 (2011), 99–101.

[3] Keith Dsouza and Mike Krebs, A combinatorial trace method: counting closed walks to assay graph eigenvalues, *Rocky Mountain J. Math.* **43** (2013), 469–478.

[4] Henry Gould, *Combinatorial Identities*, Morgantown Printing and Binding, Morgantown WV, 1972.

[5] David Guichard, Sums of selected binomial coefficients, *College Math. J.* **26** no. 3 (1995), 209–213.

[6] F. T. Howard and Richard Witt, Lacunary sums of binomial coefficients, *Applications of Fibonacci Numbers* **7** (1998), 185–195.

[7] Thomas Koshy, *Fibonacci and Lucas Numbers With Applications*, Wiley, New York, 2001.

[8] Mike Krebs and Natalie Martinez, The combinatorial trace method in action, *College Math. J.* **44** no. 1 (2013), 32–36.

[9] László Lovász, *Combinatorial Problems and Exercises* (second edition), AMS, Providence, 2007.

[10] Michael Sipser, *Introduction to the Theory of Computation* (third edition), Cengage Learning, Boston, 2013.