Analysis of Multivariate Generating Functions

Mark C. Wilson

April 15, 2005

Background

- Given a (multivariate) sequence $a_{\mathbf{n}}$ of complex numbers (usually integers) arising in a combinatorial problem, we wish to find an exact or asymptotic formula for large $|\mathbf{n}|$.
- Sequences come from areas like analysis of algorithms, computational biology, information theory, queueing theory, statistical physics.
- Sequence often gives size or statistics of large random structures.
- Typically, we apply a transform to the sequence f (generating function, Mellin transform, ...), yielding a complex function F. We then study the singularities of F and extract information on the original sequence via complex analysis.
- Computer algebra systems can automate much of the work; computations are often straightforward but horrible to do by hand.

Generating functions

- The (ordinary) generating function (GF) of the sequence $f:\mathbb{N}\to\mathbb{C}$ is the formal power series $F(z)=\sum f(n)z^n$, and element of $\mathbb{C}[[z]] := A$. We write $[z^n]F(z)=a_n$.
- In most applications F converges in a neighbourhood of the origin and defines an analytic function there. It can often be extended by analytic continuation.
- Natural operations on sequences correspond to natural operations in A. For example,
  - $[z^n]F(z)+G(z)=f(n)+g(n)$
  - $[z^n]F(z)G(z)=\sum_{j+k=n} f(j)g(k)$
  - $[z^n]zF'(z)=nf(n)$
  - $[z^n](F(z)-F(0))/z=f(n+1)$

Advantages of GFs

- Provide a compact representation of the sequence.
- Well adapted for computer algebra manipulation; symbolic computation avoids numerical errors and can yield insight.
- Can prove identities, simplify sums.
- Can solve recurrences (difference equations).
- Can yield recurrences, and hence fast computation algorithms.
- Can derive statistics routinely.
- Allow extraction of asymptotic approximations.

The GF is the best all-round tool in this area.

Toy example

- Fibonacci numbers: $a_n = a_{n-1} + a_{n-2}, a_0 = 0, a_1 = 1.$

Toy example

- Fibonacci numbers: $a_n = a_{n-1} + a_{n-2}, a_0 = 0, a_1 = 1$.
- Let $F(z) = \sum_n a_n z^n$. Then $F(z) - z = zF(z) + z^2 F(z)$ so $F(z) = z/(1 - z - z^2)$.

Toy example

- Fibonacci numbers: $a_n = a_{n-1} + a_{n-2}, a_0 = 0, a_1 = 1$.
- Let $F(z) = \sum_n a_n z^n$. Then $F(z) - z = zF(z) + z^2 F(z)$ so $F(z) = z/(1 - z - z^2)$.
- Use partial fractions to write

$$F(z) = \frac{1}{\sqrt{5}} \left(\frac{1}{1 - \phi z} - \frac{1}{1 + \phi^{-1} z} \right) \qquad \phi := \frac{1 + \sqrt{5}}{2}.$$

Toy example

- Fibonacci numbers: $a_n = a_{n-1} + a_{n-2}, a_0 = 0, a_1 = 1$.
- Let $F(z) = \sum_n a_n z^n$. Then $F(z) - z = zF(z) + z^2 F(z)$ so $F(z) = z/(1 - z - z^2)$.
- Use partial fractions to write

$$F(z) = \frac{1}{\sqrt{5}} \left(\frac{1}{1 - \phi z} - \frac{1}{1 + \phi^{-1} z} \right) \qquad \phi := \frac{1 + \sqrt{5}}{2}.$$

- Use geometric series formula (table lookup) to get

$$a_n = \frac{1}{\sqrt{5}} \left(\phi^n - (-1)^n \phi^{-n} \right) \approx \frac{1}{\sqrt{5}} \phi^n.$$

Toy example

- Fibonacci numbers: $a_n = a_{n-1} + a_{n-2}, a_0 = 0, a_1 = 1$.
- Let $F(z) = \sum_n a_n z^n$. Then $F(z) - z = zF(z) + z^2 F(z)$ so $F(z) = z/(1 - z - z^2)$.
- Use partial fractions to write

$$F(z) = \frac{1}{\sqrt{5}} \left(\frac{1}{1 - \phi z} - \frac{1}{1 + \phi^{-1} z} \right) \qquad \phi := \frac{1 + \sqrt{5}}{2}.$$

- Use geometric series formula (table lookup) to get

$$a_n = \frac{1}{\sqrt{5}} \left(\phi^n - (-1)^n \phi^{-n} \right) \approx \frac{1}{\sqrt{5}} \phi^n.$$

- Essentially, we want to automate all parts of this process, for more general recurrences/functional equations and general dimension.

The hierarchy of GFs

- Some classes of GFs occur often and are nice from the viewpoint of computational algebra.

  Rational (satisfies linear equation over $\mathbb{C}[z]$)

  $\implies$ algebraic (satisfies polynomial equation over $\mathbb{C}[z]$)

  $\implies$ holonomic (satisfies linear ODE over $\mathbb{C}[z]$).

- The first implication is trivial, the second is a theorem of Comtet (1964), and there is an algorithm to find the least degree ODE.

Recurrences yield functional equations

- Key idea: a recurrence relation on a sequence is equivalent to a functional equation satisfied by the GF.
- Standard example: the quicksort recurrence. The expected number of comparisons of (randomized) quicksort on an input permutation of size n satisfies

$$a_n = n - 1 + \frac{2}{n}\sum_{p=1}^{n-1} a_p; \qquad a_0 = 0.$$

  The GF then satisfies, by the rules above, the first order linear ODE

$$zF'(z) = 2z^2/(1-z)^3 + 2zF(z)/(1-z); \qquad F(0) = 0.$$

- F is rational iff f satisfies a linear recurrence with constant coefficients
- F is holonomic iff f satisfies a linear recurrence with polynomial coefficients.

Functional equations can yield better recurrences

- The counting GF of binary trees by internal nodes satisfies $T(z) = 1 + zT(z)^2$ (later). This is equivalent to the quadratic recurrence $a_n = \sum_{k<n} a_k a_{n-1-k}, a_0 = 1$.
- Since GF is algebraic, hence holonomic, there should be a linear recurrence.
- The answer is

$$(4z^2 - z)T'(z) + (2z - 1)T(z) + 1 = 0$$

leading to the recurrence

$$(n + 1)a_n = (4n - 2)a_{n-1} \qquad a_0 = 1.$$

This allows for much faster computation and makes it plain that a_n involves a quotient of factorials. In particular it follows that $a_n = \frac{1}{n+1}\binom{2n}{n}$, the nth Catalan number.

The symbolic method I

- Many recurrences occurring in enumeration are special cases of general constructions. The computation can be done once and for all. This leads to a formal grammar for combinatorial classes.
- A combinatorial class is a set X with a (size) function $|\cdot| : X \to \mathbb{N}$ such that for all $x \in X, n \in \mathbb{N}$, the inverse image $\{x \in X \mid |x| = n\}$ is finite.
- Let $\mathcal{A}$ be a combinatorial class, with $|\mathcal{A}_n| = a_n$. The counting GF for $\mathcal{A}$ is $A(z) = \sum_n a_n z^n = \sum_{a \in \mathcal{A}} z^{|a|}$.
- Classic examples: strings/trees/permutations/mappings with various constraints.

The symbolic method II

- Let $\mathcal{A}, \mathcal{B}$ be combinatorial classes with counting GFs $A(z), B(z)$. The size of an ordered pair of objects (α, β) is defined to be $|\alpha| + |\beta|$.
- The counting OGF of $\mathcal{A} \times \mathcal{B}$ is then

$$\sum_{\gamma \in \mathcal{A} \times \mathcal{B}} z^{|\gamma|} = \sum_{\alpha \in \mathcal{A}} \sum_{\beta \in \mathcal{B}} z^{|\alpha|+|\beta|} = \sum_{\alpha \in \mathcal{A}} z^{|\alpha|} \sum_{\beta \in \mathcal{B}} z^{|\beta|} = A(z)B(z).$$

- Also the GF for $\mathcal{A} \cup \mathcal{B}$ is $A(z) + B(z)$ if the classes are disjoint. Thus the GF for the set of sequences of elements of $\mathcal{A}$ is $1 + A(z) + A(z)^2 + \cdots = (1 - A(z))^{-1}$.
- Similar but more complicated formulae arise for sets, multisets, cycles, etc, involving $\exp, \log$ and some special infinite series.

Symbolic method example

- A binary tree is either a single external node or an internal node connected to a pair of binary trees. Let $\mathcal{T}$ be the class of binary trees:

$$\mathcal{T} = \{ext\} \cup \{int\} \times \mathcal{T} \times \mathcal{T}.$$

In terms of a formal grammar

$$< tree > = < ext > + < int > \times < tree > \times < tree > .$$

- Give $< ext >$ weight a and $< int >$ weight b to obtain $T(z) = z^a + z^b T(z)^2$. Special cases: $a = 0, b = 1$ counts trees by internal nodes; $a = 1, b = 0$ by external nodes; $a = b = 1$ by total nodes.

Substring patterns - autocorrelation polynomial

- Given arbitrary binary string $\sigma = \sigma_0\sigma_1 \cdots \sigma_{k-1}$ of length k.

Substring patterns - autocorrelation polynomial

- Given arbitrary binary string $\sigma = \sigma_0\sigma_1 \cdots \sigma_{k-1}$ of length k.
- For $0 \leq j \leq 1$, shift σ right j places. Define $c_j = 1$ if the overlap matches the tail $\sigma^{(j)}$ of σ, $c_j = 0$ otherwise. The autocorrelation polynomial is $c(z) = \sum_j c_j z^j$.

Substring patterns - autocorrelation polynomial

- Given arbitrary binary string $\sigma = \sigma_0\sigma_1 \cdots \sigma_{k-1}$ of length k.
- For $0 \leq j \leq 1$, shift σ right j places. Define $c_j = 1$ if the overlap matches the tail $\sigma^{(j)}$ of σ, $c_j = 0$ otherwise. The autocorrelation polynomial is $c(z) = \sum_j c_j z^j$.
- Let $\mathcal{S}$, (resp. $\mathcal{T}$) be the set of bitstrings not containing p (resp. containing it once at the end). Then

$$\mathcal{S} \cup \mathcal{T} \cong \{\epsilon\} \cup \mathcal{S} \times \{0, 1\}$$
$$\mathcal{S} \times \{\sigma\} \cong \mathcal{T} \times \cup_{j:c_j \neq 0}\sigma^{(j)}$$

and the symbolic method gives $S(z) + T(z) = 1 + 2zS(z)$ and $S(z)z^k = T(z)c(z)$. Thus

$$S(z) = \frac{c(z)}{z^k + (1 - 2z)c(z)}.$$

Rational GFs

- A linear recurrence relation with constant coefficients yields a rational GF and vice versa.
- Rational GFs always arise from the transfer matrix method.
- Special case: the counting GF of a regular language is rational (Chomsky-Schutzenberger, 1963). Thus if we construct a combinatorial class iteratively using only disjoint union, cartesian product, and sequence, the counting GF is rational.
- Example: let $S \subseteq \mathbb{N}$. An S-composition of n is simply a sequence with terms from S whose sum is n. If $G(z)$ is the counting GF of S then $F(z) = (1 - G(z))^{-1}$ enumerates S-compositions. Taking $S = \{1, 2, \ldots, m\}$ yields

$$F(z) = \frac{1}{1 - \frac{z(1-z^m)}{1-z}} = \frac{1 - z}{1 - 2z + z^{m+1}}.$$

- Most combinatorial examples involving counting words with certain patterns fall into this class.

Algebraic GFs

- The counting GF of an unambiguous context-free language is algebraic (Chomsky-Schutzenberger). Thus combinatorial classes constructed recursively (such as trees) using only disjoint union, cartesian product, and sequence have algebraic GFs.
- Example: binary trees have $T(z) = 1 + zT(z)^2$ as above.
- Another example: general ordered trees have $< tree >=< node > \times sequence(< tree >)$ which can be rewritten

$$\mathcal{T} = \{node\} \times \mathcal{S}$$
$$\mathcal{S} = \{\varepsilon\} \times \mathcal{S} \times \mathcal{T}$$

so that $T(z) = zS(z), S(z) = 1 + S(z)T(z)$. This polynomial system is reduced by algebraic elimination (resultants or Gröbner bases) to obtain $T(z)^2 - T(z) + z = 0$.

Software

- The MAPLE package `gfun` by Bruno Salvy, Paul Zimmermann and Eithne Murray can guess a GF from the first few terms of the sequence, find the least order ODE for an algebraic GF, convert from holonomic equation to recurrence, etc.
- Holonomic GFs can be manipulated nicely with algorithms based on Gröbner basis computations in Ore algebras. Implemented in MAPLE by F. Chyzak.
- The MAPLE package `combstruct` automates many symbolic method computations for combinatorial classes.

Similarities and differences when $d > 1$

- Given a multivariate sequence $f : \mathbb{N}^d \to \mathbb{C}$, the GF $F(\mathbf{z}) = \sum_{\mathbf{n} \in \mathbb{N}^d} f(\mathbf{n})\mathbf{z}^{\mathbf{n}}$, where $\mathbf{z}^{\mathbf{n}} = z_1^{n_1} z_2^{n_2} \dots z_d^{n_d}$ can be defined analogously.
- The basic algebra rules work as before.
- The symbolic method works as before; size is now a function into $\mathbb{C}^d$.
- The link between recurrences and functional equations is more complicated. A linear recurrence $a_{\mathbf{n}} = \sum c_{\mathbf{s}} a_{\mathbf{n}-\mathbf{s}}$ with constant coefficients may not yield a rational, algebraic, or even D-finite GF in 2 or more variables.

Strange behaviour in higher dimensions

- A linear recurrence with constant coefficients in $d \geq 2$ variables can yield a nasty GF even with nice boundary conditions. The recurrence may be “forward” in some dimensions but “forward” in others.
- Example: knight’s walk. Consider paths starting at $(1, 1)$ with jumps in $\{(2, -1), (-1, 2)\}$, staying in the first quadrant. Bousquet-Mélou and Petkovšek showed that the GF counting these is not holonomic.
- A linear recurrence with polynomial coefficients corresponds to a linear PDE. Existence and uniqueness theorems are much weaker than for ODEs.
- First order linear PDEs can be solved by the method of characteristics. While OK in theory, it is often hard to do in practice.

Results from the kernel method

- (Partially) systematized by Bousquet-Mélou and Petkovšek (2000), based on difficult work by Fayolle, Iasnogorodski, Malyshev.
- A linear recurrence $a_{\mathbf{n}} = \sum_{s \in E} c_{\mathbf{s}} a_{\mathbf{n}-\mathbf{s}}$ is given for $\mathbf{n} \geq \mathbf{q}$. If E lies on one side of a hyperplane, there is a unique solution.
- The apex of the recurrence is the coordinatewise minimum of the set $E \cup \{0\}$.
- B-M & P: if the apex has all coordinates nonnegative, then F is rational iff K is. If the apex has only one coordinate negative, then F is algebraic iff K is.
- The apex of the knight's walk has 2 negative coordinates and non-holonomic GF.

Kernel method example

- Dyck paths are enumerated by

$$a_{mn} = a_{m-1,n-1} + a_{m-1,n+1}; \qquad a_{0,m} = 0, a_{n,0} = 0, a_{00} = 1.$$

Kernel method example

- Dyck paths are enumerated by

$$a_{mn} = a_{m-1,n-1} + a_{m-1,n+1}; \qquad a_{0,m} = 0, a_{n,0} = 0, a_{00} = 1.$$

- Let $F(x, y) = \sum_{m \geq 1, n \geq 1} a_{mn} x^{m-1} y^{n-1}$. The recurrence yields $F = (K - U)/Q$ with

$$Q(x, y) = y - x - xy^2, \qquad K(x, y) = y, \qquad U(x) = \sum_{m \geq 1} a_{m1} x^m.$$

K is known (GF of boundary conditions) and U is determined by the recurrence but unknown in closed form. One equation, two unknowns!

Kernel method example

- Dyck paths are enumerated by

$$a_{mn} = a_{m-1,n-1} + a_{m-1,n+1}; \qquad a_{0,m} = 0, a_{n,0} = 0, a_{00} = 1.$$

- Let $F(x,y) = \sum_{m\geq 1, n\geq 1} a_{mn} x^{m-1} y^{n-1}$. The recurrence yields $F = (K - U)/Q$ with

$$Q(x,y) = y - x - xy^2, \qquad K(x,y) = y, \qquad U(x) = \sum_{m\geq 1} a_{m1} x^m.$$

  K is known (GF of boundary conditions) and U is determined by the recurrence but unknown in closed form. One equation, two unknowns!

- On the curve $Q = 0$, we have $y = \xi(x)$, so $U(x) = \xi(x)$ there, hence everywhere by analyticity.

Kernel method example

- Dyck paths are enumerated by

$$a_{mn} = a_{m-1,n-1} + a_{m-1,n+1}; \qquad a_{0,m} = 0, a_{n,0} = 0, a_{00} = 1.$$

- Let $F(x, y) = \sum_{m\geq 1, n\geq 1} a_{mn} x^{m-1} y^{n-1}$. The recurrence yields $F = (K - U)/Q$ with

$$Q(x, y) = y - x - xy^2, \qquad K(x, y) = y, \qquad U(x) = \sum_{m\geq 1} a_{m1} x^m.$$

  K is known (GF of boundary conditions) and U is determined by the recurrence but unknown in closed form. One equation, two unknowns!

- On the curve $Q = 0$, we have $y = \xi(x)$, so $U(x) = \xi(x)$ there, hence everywhere by analyticity.
- In fact we have by division

$$F(x, y) = \frac{\xi(x)}{1 - \xi(x)y}.$$

Trickier problems

- The kernel method with apex having a single negative coordinate is easy enough when this coordinate is -1. All examples I have found in the literature satisfy this. What happens in general?
- Example: knight's walk with $(2,1),(2,-1),(1,-2),(1,2)$ as allowed steps. By BM& P, GF is algebraic.
- Here $Q = xy^4 + x^2y^3 + y^2x^2y + x$. The branches of y near the origin must be analysed by means of Puiseux (fractional) series, Newton polygon methods, etc. An explicit useful form can be computed with substantial effort.

OK Corral example

- A model for the famous gunfight. A complete bipartite graph is given. At each time step one node is chosen at random and an edge adjacent to it. The node on the other side is deleted. Ends when one part of bipartition is eliminated.
- Recurrence for expected number of survivors is

$$a_{mn} = \frac{m}{m+n} a_{m,n-1} + \frac{n}{m+n} a_{m,n-1} \qquad a_{0,n} = n, a_{m,0} = m$$

which is equivalent to the PDE

$$[x\partial/\partial x + y\partial/\partial y] \left(F(x,y) - \frac{x}{(1-x)^2} - \frac{y}{(1-y)^2} \right) = 0.$$

- Such PDEs can be solved via the method of characteristics which reduces it to integrating 1-variable ODEs.