DIGITAL SEARCH TREES AND BASIC HYPERGEOMETRIC FUNCTIONS

Helmut Prodinger

Department of Algebra and Discrete Mathematics
Technical University of Vienna, Austria

Abstract. We use some concepts of basic hypergeometric series to give an extremely simple derivation of the exact formula for the expected number of nodes on level l in a random digital search tree, built from n random data.

1. Introduction

In this note we want to show that *basic hypergeometric functions* are the most natural concept when analyzing digital search trees. We do not give new formulæ but strikingly simple and elegant derivations for already existing ones.

In a previous paper [7], the present writer has already demonstrated the use of basic hypergeometric functions for *approximate counting*.

We only need a few things about hypergeometric functions and collect them in the next section.

2. Basics about basic hypergeometric functions

For the following we use [1]; compare also [2].

q–**Pochhammer symbol**.

$$(a)_n = (1-a)(1-aq)\dots(1-aq^{n-1}), \qquad (a)_0 = 1, \qquad (a)_\infty = \lim_{n\to\infty}(a)_n$$

We will only use this for $q=\frac12$. Nevertheless we will stay with this classic notation. In many computer science texts the notation Q_n is used for $(\frac12)_n$.

Cauchy's formula. (or: q–binomial theorem.)

$$\sum_{n\ge0}\frac{(a)_n t^n}{(q)_n} = \frac{(at)_\infty}{(t)_\infty}.$$

The special case $a=0$ is generally attributed to Euler:

$$\sum_{n\ge0}\frac{t^n}{(q)_n} = \frac{1}{(t)_\infty}.$$

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX

Now observe that

$$q^{\binom{n}{2}} = (-1)^n(0-1)(0-q)\dots(0-q^{n-1}) = (-1)^n \lim_{\varepsilon\to 0} \varepsilon^n(1/\varepsilon)_n.$$

Hence

$$\sum_{n\ge 0} \frac{q^{\binom{n}{2}} t^n}{(q)_n} = \lim_{\varepsilon\to 0} \sum_{n\ge 0} \frac{(-\varepsilon t)^n(\frac{1}{\varepsilon})_n}{(q)_n} = \lim_{\varepsilon\to 0} \frac{(-t)_\infty}{(-\varepsilon t)_\infty} = (-t)_\infty.$$

This formula is also called Euler's formula.

Heine's transformation.

$$\sum_{n\ge 0} \frac{(a)_n(b)_n t^n}{(q)_n(c)_n} = \frac{(b)_\infty(at)_\infty}{(c)_\infty(t)_\infty} \sum_{n\ge 0} \frac{(c/b)_n(t)_n b^n}{(q)_n(at)_n}.$$

Corollary.

$$\sum_{n\ge 0} (y)_n z^n = (y)_\infty \sum_{n\ge 0} \frac{y^n}{(q)_n(1-zq^n)}.$$

Proof. Set $a = q$, $b = y$, $c = 0$, and $t = z$ in Heine's formula. Then

$$\sum_{n\ge 0} (y)_n z^n = \frac{(y)_\infty(qz)_\infty}{(z)_\infty} \sum_{n\ge 0} \frac{(z)_n y^n}{(q)_n(qz)_n}.$$

Since $\dfrac{(qz)_\infty}{(z)_\infty} = \dfrac{1}{1-z}$ and $\dfrac{(z)_n}{(qz)_n} = \dfrac{1-z}{1-zq^n}$, the formula follows.

3. Level polynomials in digital search trees

A digital search tree in constructed like a binary search tree, but the decision to go down to the left or right is done according to the representation of the key as a binary string of bits. If the first bit is 0, the item goes to the left, otherwise to the right. Then the second bit is responsible for left or right, etc., until there is an empty node where the item can be stored.

The classic book [3] contains a more elaborate description.

Now, in order to study the average search costs in a digital search tree built from n random data (i.e., in every decision, 0 and 1 is equally likely), the polynomial $H_n(u)$, which has as the coefficient of u^k the expected number of nodes on this level, is studied. By convention, the root is at level 0.

The formula

$$H_n(u) = \sum_{k=1}^{n} \binom{n}{k} (-1)^{k-1}(u)_{k-1}, \qquad n \ge 1, \quad H_0(u) = 0,$$

appears already in Knuth's book [3] (compare [4]). Remember, that we set $q = \frac{1}{2}$.

Louchard [5] gave explicit expressions for the coefficients of $H_n(u)$; see also Mahmoud [6]. We write $[u^k]H_n(u)$ for the coefficient of u^k in $H_n(u)$, and similarly for generating functions in 2 variables.

Here we want to give the particularly simple derivation of this explicit formula by means of basic hypergeometric functions. It goes like this:

$$\begin{aligned}
H(u,x) := \sum_{n\ge1} H_n(u)x^n &= \sum_{k\ge1}(-1)^{k-1}(u)_{k-1}\sum_{n\ge k}\binom{n}{k}x^n \\
&= \sum_{k\ge1}(-1)^{k-1}(u)_{k-1}\frac{x^k}{(1-x)^{k+1}} \\
&= \frac{x}{(1-x)^2}\sum_{k\ge0}(u)_k\Big(\frac{x}{x-1}\Big)^k.
\end{aligned}$$

Using the corollary of Heine's transformation, we get

$$\begin{aligned}
H(u,x) &= \frac{x}{(1-x)^2}(u)_\infty\sum_{k\ge0}\frac{u^k}{(q)_k\big(1-\frac{x}{x-1}q^k\big)} \\
&= \frac{x}{1-x}(u)_\infty\sum_{k\ge0}\frac{u^k}{(q)_k\big(1-x(1-q^k)\big)} \\
&= (u)_\infty\sum_{k\ge0}\frac{u^k}{(q)_k}q^{-k}\Big[\frac{1}{1-x}-\frac{1}{1-x(1-q^k)}\Big] \\
&= \frac{(u)_\infty}{1-x}\frac{1}{(u/q)_\infty}-(u)_\infty\sum_{k\ge0}\frac{(u/q)^k}{(q)_k}\frac{1}{1-x(1-q^k)}.
\end{aligned}$$

In the last step we have used Cauchy's (or Euler's) formula. Since

$$\frac{(u)_\infty}{(u/q)_\infty}=\frac{1}{1-u/q},$$

we can already read off the coefficient of x^n:

$$[x^n]H(u,x)=\frac{1}{1-u/q}-(u)_\infty\sum_{k\ge0}\frac{(u/q)^k}{(q)_k}\big(1-q^k\big)^n.$$

Now we can also read off the coefficient of u^l:

$$\begin{aligned}
[u^lx^n]H(u,x)=[u^l]H_n(u) &= q^{-l}-\sum_{k=0}^{l}\frac{q^{-k}}{(q)_k}\big(1-q^k\big)^n[u^{l-k}](u)_\infty \\
&= q^{-l}-\sum_{k=0}^{l}\frac{q^{-k}}{(q)_k}\big(1-q^k\big)^n\frac{(-1)^{l-k}q^{\binom{l-k}{2}}}{(q)_{l-k}}.
\end{aligned}$$

In the last step we have used the second identity of Euler. This is (apart from trivial modifications) the formula (4) from [5].

The quantity

$$[u^l]\frac{1}{1-u}H_n(u)$$

was also of interest in [5] (formula (6)). It is the coefficient of u^l in

$$\frac{1}{1-u}\frac{1}{1-u/q}-(qu)_\infty\sum_{k\ge 0}\frac{(u/q)^k}{(q)_k}\big(1-q^k\big)^n.$$

Since

$$\frac{1}{1-u}\frac{1}{1-u/q}=\frac{1}{1-q}\frac{1}{1-u/q}-\frac{q}{1-q}\frac{1}{1-u},$$

we evaluate it to

$$\frac{1}{1-q}(q^n-q)-\sum_{k=0}^{l}\frac{q^{-k}}{(q)_k}\big(1-q^k\big)^n\frac{q^{l-k}(-1)^{l-k}q^{\binom{l-k}{2}}}{(q)_{l-k}}.$$

In the same style we can also deal with another variety of polynomials, the so called leaf polynomials $L_n(u)$. However, since $L_n(u)=1-(1-u/q)H_n(u)$ (see [3, 6]), the coefficients are obtained immediately from those for $H_n(u)$.

References

1. G.E. Andrews, *The Theory of Partitions*, Addison Wesley, 1976.
2. G. Gasper and M. Rahman, *Basic Hypergeometric Series*, Cambridge Unversity Press, Cambridge, 1990.
3. D.E. Knuth, *The Art of Computer Programming, Vol. 3*, Addison-Wesley, 1973.
4. A. Konheim and D. Newman, *A note on growing binary trees*, Discrete Mathematics **4** (1973), 57–63.
5. G. Louchard, *Exact and asymptotic distributions in digital and binary search trees*, RAIRO Theoretical Informatics and Applications **21** (1987), 479–495.
6. H. Mahmoud, *Evolution of Random Search Trees*, John Wiley, 1992.
7. H. Prodinger, *Approximate Counting via Euler Transform*, Math. Slovaca **44** (1994), 569–574.

TU Vienna
Wiedner Hauptstrasse 8–10
A-1040 Vienna
Austria
E-mail address: proding@rsmb.tuwien.ac.at