J. Symbolic Computation (1994) **18**, 87–90

A Computer Algebra Approach to the Discrete Dirichlet Problem

CRAIG R. ORR AND DORON ZEILBERGER

Department of Mathematics, Temple University, Philadelphia, PA 19122, USA

(Received 25 March 1994)

The Gröbner Basis Algorithm is used to find closed form expressions for the generating functions of finite difference equations. Such difference equations arise in elliptic PDE's, random walk problems, and gambler's ruin problems.

Introduction

Solutions to the Dirichlet problem for (elliptic) partial difference equations are needed both in the solutions of boundary value problems of elliptic partial differential equations (Birkhoff,1972) and in random walk problems (Doyle and Snell, 1984).

In order to approximate solutions of boundary value problems in partial differential equations, one sets up a so-called finite difference scheme on a discrete grid and considers the resulting finite difference problem. The solution gives approximations for the values of the desired function at the grid points. The finer the grid, the better the approximation. However, the effort also grows, since the number of mesh points grows.

In random walk problems, the value of interest is either the probability of getting from a fixed point to another point under a constrained walk, or the expected life, as in gambler's ruin problems. The solution can be formulated in terms of an appropriate boundary value problem for a certain region of the lattice.

As far as we know, the only method that is used for obtaining exact solutions for such discrete Dirichlet problems is by straightforward linear algebra, getting a large, but sparse, system of linear equations. A whole branch of numerical linear algebra that treats sparse systems was developed from the need to solve such boundary value problems.

Here we propose a new method for solving such problems, that reduces the problem to a much smaller system of linear equations ($O(N)$ as opposed to $O(N^2)$), albeit no longer sparse.

Our method uses a very small part of Buchberger's celebrated method of Gröbner basis, namely taking the "normal form" of a polynomial (Buchberger,1976). This method is implemented, among other places, in MAPLE in the `grobner` package.

We will illustrate our method by a detailed example: The two-dimensional gambler's ruin problem, for which no closed form solution is known to exist, and probably doesn't exist. The manipulations can be mechanized, but we put in the details for expository sake.

0747–7171/94/010087 + 04 $08.00/0

Gambler's Ruin Examples with Generating Functions

Consider a gambler who wins a dollar or loses a dollar with probability $\frac{1}{2}$. The gambler starts out with i dollars and an opponent starts out with $n-i$ dollars. The game ends when one of the two players has no money left. How long should the game be expected to last? This problem is known as the symmetric classical gambler's ruin problem. The expected duration is known and is given by $f_n(i) = i(n-i)$ (Feller, 1950, pp 311–318).

Now consider a two-dimensional version of this game: the gambler has two piles of money, say i dollars and j shekels. The opponent has $M-i$ dollars and $N-j$ shekels. In this game, the gambler wins a dollar, loses a dollar, wins a shekel, or loses a shekel each with probabiliy $\frac{1}{4}$. The game ends when either the gambler or the opponent has one pile reduced to nothing. How long should this game be expected to last?

Consider first the one-dimensional(pile) game. Let $f(i) = f_n(i)$ be the expected duration of the game. Then

$$2f(i) - f(i+1) - f(i-1) = 2.$$

Let

$$F(x) = \sum_{i=1}^{n-1} f(i)x^i$$

be the generating function for the problem. Then,

$$(2 - x - x^{-1})F(x) = 2\sum_{i=1}^{n-1} x^i - f(n-1)x^n - f(1).$$

Now let $f(n-1) = f(1) = c$. Then

$$(2 - x - x^{-1})F(x) = \frac{2(x - x^n)}{1-x} - c(x^n + 1).$$

So,

$$F(x) = \frac{(2\,x - 2\,x^n - cx^n + cx^{n+1} - c + cx)\,x}{(x-1)^3}.$$

Now F, we know, is a polynomial by definition. So it can be determined that $c = n-1$ by long division and setting the denominator equal to 0. (Alternatively, we could use l'Hopital's rule to find that $c = n-1$.) Therefore,

$$F(x) = \frac{x\left(2\,x - 2\,x^n - (n-1)\,x^n + (n-1)\,x^{n+1} - n + 1 + (n-1)\,x\right)}{(x-1)^3}.$$

Therefore, we can find $F(i)$ by finding the constant term of $F(x)/x^i$.

$$\begin{aligned} f(i) &= CT\left(-\frac{(n+1)\,x^2 - (n-1)\,x}{(1-x)^3\,x^i}\right) \\ &= (n-i)i, \end{aligned}$$

which is the expected result.

The two-dimensional(pile) case is more difficult, since we will not be able to do simple long division at the end to get the result.

Let $f(i,j) = f_{M,N}(i,j)$ be the expected duration of the game when the gambler has i dollars and j shekels. Then

$$4f(i,j) - f(i+1,j) - f(i-1,j) - f(i,j+1) - f(i,j-1) = 4$$

with

$$f(i,0) = f(i,N) = f(0,j) = f(M,j) = 0.$$

Let

$$F(x,y) = F_{M,N}(x,y) = \sum_{i=1}^{M-1}\sum_{j=1}^{N-1} f(i,j)x^i y^j.$$

Then,

$$\begin{aligned}(4 - x - x^{-1} - y - y^{-1})F(x,y) &= \frac{(4\,x - 4\,x^M)\,(y - y^N)}{(x-1)\,(y-1)} - x^M \sum_{j=1}^{N-1} f(M-1,j)y^j - \\ &\quad \sum_{j=1}^{N-1} f(1,j)y^j - y^N \sum_{i=1}^{M-1} f(i,N-1)x^i - \sum_{i=1}^{M-1} f(i,1)x^i\end{aligned}$$

Now let

$$\phi(x) = \sum_{i=1}^{M-1} f(i,N-1)x^i = \sum_{i=1}^{M-1} f(i,1)x^i$$

and

$$\psi(y) = \sum_{j=1}^{N-1} f(M-1,j)y^j = \sum_{j=1}^{N-1} f(1,j)y^j.$$

Then,

$$(4 - x - x^{-1} - y - y^{-1})F(x,y) = \frac{(4\,x - 4\,x^M)\,(y - y^N)}{(x-1)\,(y-1)} - (x^M + 1)\psi(y) - (y^N + 1)\phi(x).$$

So,

$$F(x,y) =$$

$$\frac{xy\left[4\,(x - x^M)\,(y - y^N) - (x-1)\,(y-1)\,(x^M+1)\,\psi(y) - (x-1)\,(y-1)\,(y^N+1)\,\phi(x)\right]}{(x-1)\,(y-1)\,(4xy - x^2y - xy^2 - x - y)}.$$

Now $\phi(x)$ and $\psi(y)$ are both polynomials, so let

$$\phi(x) = a_1x + a_2x^2 + \ldots + a_{M-1}x^{M-1}$$

and

$$\psi(y) = b_1y + b_2y^2 + \ldots + b_{N-1}y^{N-1}.$$

We want the numerator of $F(x,y)$ to be 0 modulo the denominator. To find the values of $\phi(x)$ and $\psi(y)$ which satisfy this condition, take the normal form of the numerator of $F(x,y)$ with respect to the (one element Gröbner basis of the) denominator. Then set

each coefficient of the reduced polynomial to 0 and solve the set of equations. From this, ϕ and ψ, and hence the generating function, can be obtained.

Applications to Other Problems

This procedure has been incorporated in a MAPLE program. Many other problems involving two-dimensional random walks can be, and have been, solved by MAPLE programs based on the above method. For example, consider an $[0, M] \times [0, N]$ grid. A particle can move north, south, east, or west with probabilities p_1, p_2, p_3, and p_4 respectively. Along the borders there are points where the particle can escape and points where the particle is captured. We wrote a MAPLE program which automatically computes the preliminary generating function for the escape probabilities, then uses Gröbner basis functions to find the true generating function. A program has also been written which deals with more general (not necessarily convex) shapes. The preliminary manipulations are almost completely mechanized. Moreover, this method could also be used for even higher dimensions, which would reduce the problem to a smaller system of equations ($O(N^{r-1})$ as opposed to $O(N^r)$ for an r-dimensional problem).

Notes

The programs mentioned above, along with others, may be obtained by e-mail from *craig@euclid.math.temple.edu* upon request.

References

Birkhoff,G. (1972).The Numerical Solutions of Elliptic Equations, *Regional Conference Series in Applied Mathematics* **1**, SIAM: Philadelphia.

Buchberger, B. (1976).Theoeretical Basis for the Reduction of Polynomials to Canonical Forms. *SIGSAM Bulletin*. **40**, 19-29.

Davenport,J.H., Siret, Y., and Tournier, E. (1988). *Computer Algebra: Systems and Algorithms for Algebraic Computation*. Academic Press: London.

Doyle, P.D., Snell, J.L. (1984). *Random Walks and Electrical Networks*. The Mathematical Association of America.

Engle, A. (1993). The Computer Solves the Three Tower Problem, *American Math. Monthly* **100**, 62–64.

Feller, W.D. (1950). *An Introduction to Probability Theory and Its Applications*, New York: John Wiley and Sons.

Zeilberger, D. (1991). The Method of Creative Telescoping, *J. Symbolic Computation* **11**, 195–204.