### A Probabilistic Remark on Algebraic Program Testing

**Title:** A Probabilistic Remark on Algebraic Program Testing  
**Authors:** Richard A. DeMillo and Richard J. Lipton

**Abstract:** A key step in Howden's method (Howden, W. E., "Algebraic Program Testing" Computer Science Technical Report No. 14, November 1976, UC-San Diego, La Jolla, CA.) for algebraic program testing requires checking the algebraic identity of multinomials. Howden's solution requires evaluations in at least $2^m$ points for $m$-ary multinomials. This note presents a probabilistic solution which achieves small probability of error on 30 points.

**Keywords:** Program Testing, Computer programs, Multinomials, Computer systems programs, Algebra, Probability.
A PROBABILISTIC REMARK ON
ALGEBRAIC PROGRAM TESTING

Richard A. DeMillo
School of Information and Computer Science
Georgia Institute of Technology
Atlanta, GA 30332

Richard J. Lipton
Computer Science Department
Yale University
New Haven, CT 06520
ABSTRACT: A key step in Howden's method [5] for algebraic program testing requires checking the algebraic identity of multinomials. Howden's solution requires evaluations in at least $2^m$ points for m-ary multinomials. This note presents a probabilistic solution which achieves small probability of error on 30 points.
Until very recently, research in software reliability has divided quite neatly into two -- usually warring -- camps: methodologies with a mathematical basis and methodologies without such a basis. In the former view, "reliability" is identified with "correctness" and the principle tool has been formal and informal verification [1]. In the latter view, "reliability" is taken to mean the ability to meet overall functional goals to within some predefined limits [2,3]. We have argued in [4] that the latter view holds a great deal of promise for further development at both the practical and analytical levels. Howden [5] proposes a first step in this direction by describing a method for "testing" a certain restricted class of programs whose behavior can -- in a sense Howden makes precise -- be algebraicized. In this way, "testing" a program is reduced to an equivalence test, the major components of which become

(i) a combinatorial identification of "equivalent" structures;

(ii) an algebraic test

\[ f_1 \equiv f_2, \]

where \( f_i, i = 1, 2 \) is a multivariable polynomial (multinomial) of degree specified by the program being considered.

In arriving at a method for exact solution of (ii), Howden derives an algorithm which requires evaluation of multinomials \( f(x_1, \ldots, x_m) \) of maximal degree \( d \) at \( O(d + 1)^m \) points. For large values of \( m \) (a typical case for realistic examples), this method becomes prohibitively expensive.

Since, however, a test for reliability rather than a certification of correctness is desired, a natural question is whether or not Howden's method can be improved by settling for less than an exact solution to (ii).
We are inspired by Rabin [6] and, less directly, by the many successes of Erdős and Spencer [7] to attempt a probabilistic solution to (ii). Using these methods, we show that (ii) can be tested with probability of error $\varepsilon$ with only $O(g(\varepsilon))$ evaluations of multinomials, where $g$ is a slowly growing function of only $\varepsilon$. In particular, 30 or so evaluations should give sufficiently small probability of error for most practical situations. The remainder of this note is devoted to proving this result.

Let us denote by $P^0(m,d)$ the class of multinomials

$$f(x_1, \ldots, x_m) \neq 0$$

(over some arbitrary but fixed integral domain) whose degree does not exceed $d > 0$. We define

$$P(m,d,r) = \min_{f \in P^0(m,d)} \operatorname{Prob}\{1 \leq x_1 \leq r, f(x_1, \ldots, x_m) \neq 0\}$$

We think of $P(m,d,r)$ as the minimal relative frequency with which witnesses to the non-nullity of a multinomial of the appropriate kind can occur in the chosen interval. Notice, in particular, that since a polynomial of degree $d$ has at most $d$ roots (ignoring multiplicity), the largest probability of finding a root must be at least the probability of finding a root by randomly sampling in the interval $1 \leq x_1 \leq r$; thus

$$P(1,d,r) \geq 1 - d/r.$$ 

Now, consider some

$$f(x_1, \ldots, x_m, y) \neq 0$$

of degree at most $d$. But there are then multinomials $\{g_i\}_{i \leq d}$, not all $\neq 0,$
such that
\[ f(x_1, \ldots, x_m, y) = \sum_{i=0}^{d} g_i(x_1, \ldots, x_m) y^i. \]

Let us suppose that \( g_k \in P_0(m, d) \). Thus
\[ \text{Prob} \{1 \leq x_1 \leq r, f(x_1, \ldots, x_m) \neq 0\} \]
\[ \geq \text{Prob} \{g_k(x_1, \ldots, x_m) \neq 0 \text{ and } y \text{ is not a root}\} \]
\[ \geq P(m, d, r)(1 - d/r). \]

Continuing inductively, we obtain
\[ P(m, d, r) \geq (1 - d/r)^m \quad (1) \]

But
\[ \lim_{m \to \infty} (1 - d/r)^m = \lim_{m \to \infty} \left[ 1 + \frac{1}{m} \left( \frac{-dm}{r} \right) \right]^m = e^{-\frac{dm}{r}}. \quad (2) \]

Combining (1) and (2), we have for large \( m, r = dm \),
\[ P(m, d, dm) \geq e^{-1}. \]

Thus, with \( t \) evaluations of \( f \) for independent choices of points from the \( m \)-cube with sides \( r = dm \), the probability of missing a witness to the non-nullity of \( f(x_1, \ldots, x_m) \) is at most
\[ (1 - e^{-1})^t. \]
Table 1 shows the probable error in testing \( f \equiv 0 \) by \( t \) evaluations of \( f \) at randomly chosen points for some typical values of \( d,m,r,t \).

\[
[1 - P(m,d,r)]^t
\]

<table>
<thead>
<tr>
<th>( dm )</th>
<th>( r )</th>
<th>( t=10 )</th>
<th>( t=20 )</th>
<th>( t=30 )</th>
<th>( t=50 )</th>
<th>( t=100 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>10</td>
<td>( 1.0 \times 10^{-2} )</td>
<td>( 1.0 \times 10^{-3} )</td>
<td>( 1.0 \times 10^{-6} )</td>
<td>( 1.1 \times 10^{-10} )</td>
<td>( 1.2 \times 10^{-20} )</td>
</tr>
<tr>
<td>20</td>
<td>10</td>
<td>0.23</td>
<td>5.5 ( \times 10^{-2} )</td>
<td>1.3 ( \times 10^{-2} )</td>
<td>7.0 ( \times 10^{-4} )</td>
<td>4.8 ( \times 10^{-7} )</td>
</tr>
<tr>
<td>50</td>
<td>10</td>
<td>0.93</td>
<td>0.87</td>
<td>0.82</td>
<td>0.71</td>
<td>0.51</td>
</tr>
<tr>
<td>( 10^2 )</td>
<td>10</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>10</td>
<td>( 10^2 )</td>
<td>( 6.0 \times 10^{-9} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
<tr>
<td>20</td>
<td>( 10^2 )</td>
<td>( 3.9 \times 10^{-8} )</td>
<td>( 1.5 \times 10^{-15} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
<tr>
<td>50</td>
<td>( 10^2 )</td>
<td>( 8.9 \times 10^{-5} )</td>
<td>( 7.9 \times 10^{-9} )</td>
<td>( 7.0 \times 10^{-13} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
<tr>
<td>( 10^3 )</td>
<td>( 10^2 )</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>10</td>
<td>( 10^3 )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
<tr>
<td>20</td>
<td>( 10^3 )</td>
<td>( 9.3 \times 10^{-18} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
<tr>
<td>50</td>
<td>( 10^3 )</td>
<td>( 7.6 \times 10^{-14} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
<td>(&lt;10^{-20} )</td>
</tr>
</tbody>
</table>

Table 1. Probable Error in Testing \( f(x_1, \ldots, x_m) = 0 \)
(degree \( \leq d \)) by \( t \) random evaluations in \( \{1, \ldots, r\} \)

Notice that for \( dm = r \), \( t = 30 \), this is already \( <10^{-5} \).
References


