Comments on "Design of Diagnosable Iterative Arrays"

A. D. FRIEDMAN AND P. R. MENON

Abstract—This correspondence shows that, contrary to a statement in the cited paper, fault location is sometimes possible in one-dimensional arrays in which only the outputs of the rightmost cell are observable.

Index Terms—Fault location, iterative arrays, one-dimensional.

A recent paper by Landgraf and Yau (and also an earlier paper by Kautz) states that fault location is impossible in one-dimensional arrays without observable cell outputs (1), since an error detected in the output of the rightmost cell can be produced by a fault in any cell of the array. Although the faulty cell itself cannot be located under the assumption of these papers, the fault can sometimes be located under a two-cell block.

We shall use the same notation and assumptions as in the earlier papers (1). The cells in the array are identical; each cell contains only combinational logic and only one cell in the array may be faulty. A fault in a cell may affect its outputs in any arbitrary manner. A typical cell in the array receives an input x from its left-hand neighbor and an external input z, and transmits an output y to its right-hand neighbor. The x input to the first cell and the y inputs to all cells can be controlled independently, and the only observable output is the z output of the rightmost cell.

Necessary and sufficient conditions for fault location to k-cell blocks in one-dimensional arrays without z outputs are presented in an earlier paper (2). The following example demonstrates that fault location is sometimes possible even when the cells have no z outputs.

Consider the following flow table, which specifies the z output of a cell for every input combination.

<table>
<thead>
<tr>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>x4</th>
</tr>
</thead>
<tbody>
<tr>
<td>z1</td>
<td>x2</td>
<td>x3</td>
<td>z2</td>
</tr>
<tr>
<td>z3</td>
<td>x1</td>
<td>x3</td>
<td>z2</td>
</tr>
</tbody>
</table>

Any one-dimensional array of these cells is testable. Let us assume that an observed incorrect output could have been produced if the first cell in the array produced an output of x1 instead of x1. Since the z input to the second cell is known, we can determine the correct output of the second cell. By changing the z input to the second cell to z3 or x4, we can guarantee that its output will be correct whether or not cell 1 is faulty. If the observable z output of the rightmost cell is now correct, this implies that the fault must have been in the first or second cell. If the observable output is still faulty, the fault cannot be in the first cell (since it would have been "masked" in the second cell). If the first cell is fault-free, the possible fault in the second cell can be masked in the third cell, and so on until the fault is located to a two-cell block. This technique also leads to a method of realizing any flow table to permit fault location.

References


Comments on "A Fast Digital Computer Method for Recursive Estimation of the Mean"

J. C. MAJITHIA

Abstract—Several algorithms for the computation of the mean value of a random variable have been analyzed. Results have shown that the two previously published techniques and a modified version of one of these, give unbiased estimates of the mean value. Furthermore, although the confidence intervals have been found to be increased in all cases, the actual increase is insignificant, while the possible speed improvements in computation can be considerable.

Index Terms—Confidence intervals, mean value, recursive estimation, unbiased estimation.

In the above correspondence, a method for computation of the mean value of a random variable uses the recursive definition of the mean value, i.e., after the nth sample, the mean value is given by

\[ \hat{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i \]

where \( \hat{X}_n \) is the nth sample.

In order to increase the speed of computation, the division operation is replaced by a shift operation. This requires approximation of n by \( M_n \), which is an integer power of 2, resulting in a modified estimate of the mean value

\[ \tilde{X}_n = \frac{1}{n} \sum_{i=1}^{n} x_i \]

where the \( x_i \) coefficients are governed by the choice of \( M_n \).

For finite sample size n, the sample mean \( \bar{X} \) has a normal distribution with mean and variance given by,

\[ E[\hat{X}_n] = \mu \]

and

\[ Var(\hat{X}_n) = \frac{\sigma^2}{n} \]

In order to obtain an unbiased estimate of the mean, it is necessary that \( E[\tilde{X}_n] = \mu \). Thus two criteria are required to investigate the efficiency of a particular recursive estimation algorithm.

1) Test \( \alpha \). Ideally if \( \alpha = 1 \), an unbiased estimate is obtained.
2) The standard deviation \( \sigma(X) = \alpha \sqrt{n} \) (based on \( \alpha = 1/n \) for all \( i \)) establishes the basic criterion for the measured mean values. The modified algorithm gives \( \sigma(\tilde{X}) = \beta \sigma \). The statistical efficiency related to confidence interval is expressed by the percentage difference \( E \), where

\[ E = \frac{\sqrt{n} - \beta^{-1}}{\sqrt{n}} \times 100 \%

If \( E = 0 \) then we have the ideal condition. In practice, however, this is not the case as has already been shown. Three methods of choosing \( M_n \) have been investigated. These are the following.

Algorithm 1: For a given n, a number \( M_n \) is chosen such that

\[ E = 0 \]

 Manuscript received September 9, 1971.

Comments on "A Fast Digital Computer Method for Recursive Estimation of the Mean"

J. C. MAJITHIA

Abstract—Several algorithms for the computation of the mean value of a random variable have been analyzed. Results have shown that the two previously published techniques and a modified version of one of these, give unbiased estimates of the mean value. Furthermore, although the confidence intervals have been found to be increased in all cases, the actual increase is insignificant, while the possible speed improvements in computation can be considerable.

Index Terms—Confidence intervals, mean value, recursive estimation, unbiased estimation.

In the above correspondence, a method for computation of the mean value of a random variable uses the recursive definition of the mean value, i.e., after the nth sample, the mean value is given by

\[ \hat{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i \]

where \( \hat{X}_n \) is the nth sample.

In order to increase the speed of computation, the division operation is replaced by a shift operation. This requires approximation of n by \( M_n \), which is an integer power of 2, resulting in a modified estimate of the mean value

\[ \tilde{X}_n = \frac{1}{n} \sum_{i=1}^{n} x_i \]

where the \( x_i \) coefficients are governed by the choice of \( M_n \).

For finite sample size n, the sample mean \( \bar{X} \) has a normal distribution with mean and variance given by,

\[ E[\hat{X}_n] = \mu \]

and

\[ Var(\hat{X}_n) = \frac{\sigma^2}{n} \]

In order to obtain an unbiased estimate of the mean, it is necessary that \( E[\tilde{X}_n] = \mu \). Thus two criteria are required to investigate the efficiency of a particular recursive estimation algorithm.

1) Test \( \alpha \). Ideally if \( \alpha = 1 \), an unbiased estimate is obtained.
2) The standard deviation \( \sigma(X) = \alpha \sqrt{n} \) (based on \( \alpha = 1/n \) for all \( i \)) establishes the basic criterion for the measured mean values. The modified algorithm gives \( \sigma(\tilde{X}) = \beta \sigma \). The statistical efficiency related to confidence interval is expressed by the percentage difference \( E \), where

\[ E = \frac{\sqrt{n} - \beta^{-1}}{\sqrt{n}} \times 100 \%

If \( E = 0 \) then we have the ideal condition. In practice, however, this is not the case as has already been shown. Three methods of choosing \( M_n \) have been investigated. These are the following.

Algorithm 1: For a given n, a number \( M_n \) is chosen such that

\[ E = 0 \]

 Manuscript received September 9, 1971.
subject to the constraint that $2^{r-1} < n \leq 2^r$.

Algorithm 2: Given $n$, choose a number $N_n$ such that
\[ N_n = 2^r \geq n \]
or
\[ N_n = 2^s < n \]
and subject to constraint that $|2^s - n| \leq 2^{r-1}$.

Algorithm 3: A modified version of Algorithm 1 using the fact that when $n$ is an integer power of 2 no approximation is necessary. Thus we use $M_n$ as given by Algorithm 1 everywhere in the region $2^{r-1} < n < 2^r$. However, when $n$ is an integer power of 2, the exact formula is used, i.e., if $n=2^r$ then
\[ \hat{X}_n = \frac{1}{2^{2r-1}} \sum_{\ell=1}^{2^{2r-1}} X_{\ell}. \]

This exact value is then used in the recursive formula to compute mean values at other sample sizes in the range $2^{r-1} < n < 2^r$. Note that it is also possible to use a similar modified version of Algorithm 2. However, the computation time in this case will be greater than for the modified version of Algorithm 1. The $E$ values computed for the three techniques show that at small values of $n$, Algorithm 3 is more efficient than either Algorithms 1 or 2. For large sample sizes it is as efficient as Algorithm 2. It does, however, have the advantage that it is as efficient as the most efficient algorithm at sample sizes which are integer powers of 2. For large sample sizes ($n > 500$) both Algorithms 2 and 3 have a maximum $E$ value of less than 3 percent. This implies that the confidence interval is increased by about 3 percent which is insignificant.

An interesting feature of the three algorithms is that the $a$ values are exactly equal to unity, for all sample sizes, resulting in unbiased estimates of the mean value.


Table I: Example of Algorithm for Walsh Function Evaluation

<table>
<thead>
<tr>
<th>$22_{16}$</th>
<th>Binary notation for wal $(22, \theta)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>101101</td>
<td>Gray code for 22</td>
</tr>
<tr>
<td>11101</td>
<td>Bit-reversed Gray code</td>
</tr>
<tr>
<td>10111 $J$</td>
<td>Bit notation for $\theta$</td>
</tr>
<tr>
<td>0.34375 $M$</td>
<td>Bits of $K$ enabled by $J$</td>
</tr>
<tr>
<td>0.0011 $M$</td>
<td>Parity check for even number</td>
</tr>
<tr>
<td>1</td>
<td>of one's in $M$</td>
</tr>
<tr>
<td>1</td>
<td>Value of wal $(22, 0.34375)$</td>
</tr>
</tbody>
</table>

A nonsingular matrix $A$ is defined as simply invertible if it can be additively decomposed as the sum of a lower triangular matrix $L$ and an upper triangular matrix $U$ such that
\[ A = L + U \]
and
\[ A^{-1} = L^{-1} + U^{-1}. \]

From (1) and (2) we obtain
\[ (L + U)(L^{-1} + U^{-1}) = I. \]

Setting
\[ K = LU^{-1} \]
we get
\[ K^2 + K + I = 0. \]

Thus the class of SIM is derivable from $K = LU^{-1}$ satisfying (5); since $K$ of order $n$, in addition, has to satisfy its characteristic equation of degree $n$ (Cayley–Hamilton theorem),
\[ K^n + \alpha_{n-1} K^{n-1} + \cdots + \alpha_1 K + \alpha_0 I = 0, \]

(5) should be the minimal equation of $K^2$ for $n > 2$.

Manuscript received August 23, 1971; revised September 21, 1971.

The authors are with the Department of Electrical Engineering, McMaster University, Hamilton, Ont., Canada.


Manuscript received June 29, 1971; revised September 21, 1971.

The authors are with the Department of Applied Mathematics, Indian Institute of Science, Bangalore-12, India.