CS 122

HEUR ISTIC ANALYS IS OF NUMER ICAL VAR IANTS OF THE GRAM-SCHMIDT ORTHONORMALIZATION PROCESS

BY

W. C. MITCHELL
AND
D. L. MCCRAITH

TECHNICAL REPORT NO. CS 122
FEBRUARY 24, 1969

COMPUTER SC IENCE DEPARTMENT
School of Humanities and Sciences
STANFORD UNIVERSITY

HEURISTIC ANALYSIS OF NUMERICAL VARIANTS OF THE GRAM-SCHMIDT ORTHONORMALIZATION PROCESS

W. C. Mitchell

and

D. L. McCraith

Department of Computer Science

Stanford University

Stanford, California

* This work was supported in part by the National Science Foundation and the Office of Naval Research.

Acknowledgements

The authors would like to express their thanks to Dr. George Forsythe of Stanford University for suggestions and criticisms of this paper.

D.L. McCraith's current address is:

Lincoln Labs
Massachusetts Institute of Technology
Lexington, Massachusetts

HEURISTIC ANALYSIS OF NUMERICAL VARIANTS OF THE GRAM-SCHMIDT ORTHONORMALIZATION PROCESS

by

W.C. Mitchell

and

D.L. McCraith

Department of Computer Science
Stanford University
Stanford, California

1. Introduction. One of the fundamental formulas of analysis is the Gram-Schmidt, orthonormalization process. Unfortunately it is also notoriously computationally unstable. Rice [1] presents some computational techniques which seem to reduce the numerical error propagation, but he presents no analysis explaining why his methods work. This paper attempts to provide a heuristic error analysis of the Gram-Schmidt process which will show why it is unstable and why Rice's techniques reduce numerical error.

2. Gram-Schmidt Process. This section will present a basic exposition of the Gram-Schmidt process showing the principal sources of round-off error.

Consider a set of linearly independent vectors $W_1, W_2 \ldots W_N$ in E_N, with an inner-product $(\cdot\ ,\ \cdot)$. We want to find a set of orthonormal vectors $V_1, V_2, \ldots V_N$ such that, for each i, V_i is a linear combination of $W_1, W_2, \ldots W_i$. The Gram-Schmidt process does this in a straight forward manner as follows:

* The authors would like to acknowledge the financial support of the National Science Foundation while both were at the Department of Computer Science, Stanford University and the assistance of the Office of Naval Research in the preparation of this report for publication.

$$(1)\qquad \left.\begin{aligned} U_n &= W_n - \sum_{i=1}^{n-1} (W_n, V_i)\ V_1 \\ V_n &= U_n / \|U_n\| \end{aligned}\right\} \quad (n=1,\ 2,\ \ldots.\ N)$$

$$\text{where } \|Z\| = (Z,\ Z)^{\frac{1}{2}}.$$

If we define $\alpha_n = 1/\|U_n\|$ and $k_{ni} = \alpha_n (W_n, V_i)$ then (1) becomes,

$$(2)\qquad V_n = \alpha_n W_n - \sum_{i=1}^{n-1} k_{ni} V_i.$$

If we denote by the vector ξ_n the numerical round-off error in evaluating (2) in finite-precision floating-point arithmetic on a computer, then (2) becomes,

$$(3)\qquad V_n = \alpha_n W_n - \sum_{i=1}^{n-1} k_{ni} V_i + \xi_n.$$

If we orthonormality as the measure of the error in a set of computed vectors, then we are interested in the magnitues of,

$$\varepsilon_{nm} = (V_n, V_m) - \delta_{nm}.$$

In the absence of round-off error $\varepsilon_{nm} = 0$ for $1 \leq n \leq N$, $1 \leq m \leq N$.

3. Heuristic Error Analysis. Since the objective of this paper is to present a heuristic analysis rather than to establish rigorous error bounds, we shall make a number of assumptions about the error terms. The validity of these assumptions will be supported by numerical experiments.

In this type of heuristic error analysis some notation for the numerical size of a quantity is needed. The $0(\tau)$ notation is too precise a concept

for this type of analysis. Dr. George Forsythe has informally proposed the notation $Y = \theta(\tau)$ to mean $Y = \theta\tau$, where $|\theta| < K$ for some unknown constant K with the general assumption that $K < 10^2$. (The value 10^2 is subject to change as needed).

Since we are interested in stable numerical procedures, we assume that $|\varepsilon_{n\pm k}| \leq .01$, and it is understood that the error analysis will be abandoned once this limit is exceeded. This assumption allows us to do a first order error analysis.

Consider the general Gram-Schmidt process (1). Since the normalization of U_n is the last arithmetic operation performed, previous errors do not directly affect (V_n,V_n). Thus we may assume. that $(V_n,V_n) = 1 + \theta(\tau)$ where $\tau = \frac{1}{2}\beta^{1-t}$ for a machine with t digits in β radix. Now, if all errors associated with V_m were of magnitude τ, we would have a numerical process as accurate as we could reasonably expect. Since the Gram-Schmidt method is not such a process we can assume that ε_{nn} is inconsquential in comparison with other errors associated with V_n. We can thus ignore ε_{nn} in a first order error analysis.

Since $\varepsilon_{nm} = \varepsilon_{mn}$, and in light of the previous assumption that $\varepsilon_{nn} = 0$, it will be convenient to consider ε_{nm} for $m<n$ unless specified otherwise.

Since we are interested in the growth of round-off error we will occasionally assume that ε_{nm} grows with n. Then, for a first order error analysis, we may ignore ε_{pm} terms in comparison with ε_{nm} terms when $p<n$.

Consider the basic round-off error vector ξ_n in (3). Since n-1 vector subtractions, n inner-product evaluations, and n multiplications of a vector by a scalar are required to compute V_n, it is reasonable to assume $\|\xi_n\| = \theta(n\tau)$. Expanding V_n as in (3) we can derive an error propagation formula

for the basic Gram-Schmidt method:

$$(4)\qquad \varepsilon_{nm} = (V_n, V_m)$$

$$= \alpha_n(W_n, V_m) - \sum_{i=1}^{n-1} k_{ni}(V_i, V_m) + (\xi_n, V_m)$$

$$= \alpha_n(W_n, V_m) - \alpha_n(W_n, V_m)(V_m, V_m)$$

$$- \sum_{i=1}^{m-1} k_{ni}\varepsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im}$$

$$+ (\xi_n, V_m)$$

$$= \alpha_n(W_n, V_m)(1 - (1-\varepsilon_{mm}))$$

$$- \sum_{i=1}^{m-1} k_{ni}\varepsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im}$$

$$+ (\xi_n, V_m)$$

$$= -\sum_{i=1}^{m-1} k_{ni}\varepsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im} + \theta(n\tau)$$

because $\varepsilon_{mm} = 0$.

Since we are interested in errors larger than τ, we will often ignore the $\theta(n\tau)$ term in (4) in the presence of other error terms. Thus (4) will often be used in the following form,

$$(5)\qquad \varepsilon_{nm} = -\sum_{i=1}^{m-1} k_{ni}\varepsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im}$$

$$\varepsilon_{2,1} = \theta(2\tau).$$

4. Modified Gram-Schmidt Process. Rice [1] has proposed a simple modification of the standard Gram-Schmidt process which seems to reduce severe error propagation, though he presents no analysis to explain the experimental success of the method. It involves no additional computation, though the definition is a bit more involved:

$$(6) \qquad U_n^1 = W_n$$

$$U_n^i = U_n^{i-1} - (U_n^i, V_i)\, V_i \qquad (i=1,2,\ldots,n-1)$$

$$V_n = U_n^n / \| U_n^n \|$$

or

$$U_n^n = W_n - \sum_{i=1}^{n-1} (U_n^i, V_i) V_i .$$

To further illustrate the relationship between this process and the regular Gram-Schmidt method; (1) could be written in the form of (6) as follows:

$$U_n^1 = W_n$$

$$U_n^i = U_n^1 - (W_n, V_i) V_i$$

$$V_n = U_n^n / \| U_n^n \| .$$

If we define $\alpha_n^* = 1/\| U_n^n \|$, $V_n^i = \alpha_n^* U_n^1$, and $k_{ni}^* = \alpha_n^* (U_n^i, V_i) = (V_n^i, V_i)$ then (6) becomes,

(7) $$V_n^1 = \alpha_n^* W_n$$

$$V_n^{i+1} = V_n^i - k_{ni}^* V_i$$

$$V_n = V_n^n.$$

An error analysis of this scheme depends on the following lemma, which is easily proven by backwards induction on m:

(8) $$V_n = V_n^m - \sum_{i=m}^{n-1} k_{ni}^* V_i.$$

proof: The lemma holds trivially for m=n. Assume it holds for m=p+1. Then,

$$\begin{aligned} V_n &= V_n^{p+1} - \sum_{i=p+1}^{n-1} k_{ni}^* V_i \\ &= V_n^p - k_{np}^* V_p - \sum_{i=p+1}^{n-1} k_{ni}^* V_i \\ &= V_n^p - \sum_{i=p}^{n-1} k_{ni}^* V_i. \end{aligned}$$

This completes the proof of (8).

From (8) we get an error formula for computation:

$$\begin{aligned} \varepsilon_{nm} &= (V_n, V_m) \\ &= (V_n^m, V_m) - \sum_{i=m}^{n-1} k_{ni}^* (V_i, V_m) + (\xi_n, V_m) \\ &= \alpha_n^* (U_n^m, V_m) - \alpha_n^* (U_n^m, V_m)(V_m, V_m) \\ &\quad - \sum_{i=m+1}^{n-1} k_{ni}^* \varepsilon_{im} + (\xi_n, V_m). \end{aligned}$$

Making the same assumptions regarding (V_m, V_m) that we made before, we get,

$$(9) \qquad \varepsilon_{nm} = - \sum_{i=m+1}^{n-1} k^{*}_{ni} \varepsilon_{im} + \theta(n\tau).$$

The improvement of (9) over (4) is obvious.

Having devised an error formula (9) for the modified Gram-Schmidt process (6) which is similar to that of the regular process (5), it is appropriate to consider the relation between k_{ni} and k^{*}_{ni}. A consideration of the properties of orthonormal vectors shows that they would be identical if there were no round-off error. This is an important consideration for the theoretical properties of the two methods, but does not explain why the Modified Gram-Schmidt process seems to be better. From (7) it is apparent that,

$$V^{i}_{n} = \alpha^{*}_{n} W_{n} - \sum_{j=1}^{i-1} k^{*}_{nj} V_{j}.$$

From this we get,

$$(10) \qquad k^{*}_{ni} = (V^{i}_{n}, V_{i})$$

$$= \alpha^{*}_{n} (W_{n}, V_{i}) - \sum_{j=1}^{i-1} k^{*}_{nj} (V_{j}, V_{i})$$

$$= \frac{\alpha^{*}_{n}}{\alpha_{n}} k_{ni} - \sum_{j=1}^{i-1} k^{*}_{nj} \varepsilon_{ij}.$$

It is thus apparent that k_{ni} and k^{*}_{ni} are of similar magnitude until the error becomes truly severe. Since we are conducting a first-order error analysis with the assumption that $|\varepsilon_{ij}| < .01$, we may assume $k^{*}_{ni} = k_{ni}$ for the purposes of our analysis.

Recall that for the regular Gram-Schmidt process we had,

$$(4) \qquad \varepsilon_{nm} = -\sum_{i=1}^{m-1} k_{ni}\varepsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im} + \theta(n\tau).$$

Since we are interested in the conditions under which severe error propagation occurs, it is reasonable to assume that ε_{nm} increases with n (since $m < n$). Then, assuming that the k_{ni} are of similar magnitude for $i < n$ and for $i < n$, we can assume error bounds for the regular Gram-Schmidt process as follows:

$$(11) \qquad \varepsilon_{nm} = \sum_{i=m+1}^{n-1} k_{ni}\varepsilon_{im} + \theta(\varepsilon_{m,m-1}) \quad (m < n - 1)$$

$$\varepsilon_{n,n-1} = -\sum_{i=1}^{n-2} k_{ni}\varepsilon_{n-1,i} + \theta(n).$$

The simularity of this and (9) allows us to use the same error analysis for both methods, the only difference being for $\varepsilon_{n,n-1}$.

5. Error Propagation Analysis. For the purpose of further analysis we will assume that $|k_{ni}| \leq K_n$ independently of i. We will also define $e_{nm} = |\varepsilon_{nm}|$. The error propagation- formulas then become:

$$(12) \qquad e_{nm} \leq K_n \sum_{i=m+1}^{n-1} e_{im} \qquad (m < n - 1)$$

$$e_{n,n-1} \leq \begin{cases} K_n \sum_{i=1}^{n-2} e_{n-1,i} & \text{Regular Gram-Schmidt} \\ \theta(n\tau). & \text{Modified Gram-Schmidt} \end{cases}$$

The following lemma is required for this analysis:

$$(13) \qquad 1 + \sum_{i=p}^{q} b_i \prod_{j=p}^{i-1} (b_j + 1) = \prod_{j=p}^{q} (b_j + 1).$$

proof: For $q < p$, $1 + 0 = 1$. Now by induction on q for $p \leq q$:

$$1 + \sum_{i=p}^{q+1} bi \prod_{j=p}^{i-1} (b_j + 1) = b_{q+1} \prod_{j=p}^{q} (b_j + 1)$$

$$+ 1 + \sum_{i=p}^{q} b_i \prod_{j=p}^{i-1} (b_j + 1)$$

$$= b_{q+1} \prod_{j=p}^{q} (b_j + 1)$$

$$+ \prod_{j=p}^{q} (b_j + 1)$$

$$= \prod_{j=p}^{q+1} (b_j + 1).$$

This completes the proof of the lemma.

We can now show that for $m < n-1$,

$$(14) \qquad e_{nm} \leq K_n e_{m+1,m} \left(\prod_{i=m+2}^{n-1} (K_i + 1) \right).$$

proof: By induction on n for $n \geq m+2$:

for $n=m+2$, (12) gives,

$$e_{nm} \leq K_n e_{n-1,m} \leq K_{m+2}, \leq K_n e_{m+1,m} \prod_{i=m+2}^{n-1} (K_i + 1).$$

For $n > m+2$ by induction on n:

$$e_{n+1,m} \leq K_{n+1} \sum_{i=m+1}^{n} e_{im}$$

$$\leq K_{n+1}(e_{m+1,m} + \sum_{i=m+2}^{n} K_i e_{m+1,m} \prod_{j=m+2}^{i-1}(K_j + 1))$$

$$\leq K_{n+1} e_{m+1,m} (1 + \sum_{i=m+2}^{n} K_i \prod_{j=m+2}^{i-1} (K_j + 1))$$

$$\leq K_{n+1} e_{m+1,m} \prod_{i=m+2}^{n} (K_i + 1)$$

which completes the proof. The final step used the lemma with $p=m+2$ and $q=n$.

This completes the error analysis for the modified Gram-Schmidt process. In summary,

$$e_{n\,m} \leq K_n \prod_{i=m+2}^{n-1} (K_i + 1) \cdot \theta(n\tau). \tag{15}$$

This is maximized for a given n by m = 1. This is verified by computational results. Table 2 presents the ε_{nm} for a modified Gram-Schmidt procedure with $k_{ni} = -9$. The results are in good agreement with (15). Indeed, it appears that the error bounds for e_{nm} are achieved by the ε_{nm}. This reflects the fact that the above proof can also be used to show the following:

(16) Defining $K_n = -k_{ni}$ independently of i,

$$\varepsilon_{nm} = K_n \varepsilon_{m+1,m} \left(\prod_{i=m+2}^{n-1} (K_i + 1) \right)$$

$$= K_n \left(\prod_{i=m+2}^{n-1} (K_i + 1) \right) \cdot \theta(n\tau).$$

For an analysis of the regular Gram-Schmidt procedure we replace (14) with,

$$e_{nm} \leq (K_n + 1)\, e_{m+1,m} \left(\prod_{i=m+2}^{n-1} (K_i + 1) \right)$$

$$\leq e_{m+1,m} \prod_{i=m+2}^{n} L_i \cdot$$

where we define $L_i = K_i + 1$. With this we can prove that for $m \geq 2$,

(17) $$e_{m+1,m} \leq 2^{m-2} e_{2,1} \prod_{i=3}^{m+1} (K_i + 1).$$

proof: By induction on $m \geq 2$.

For m = 2, from (11) $e_{3,2} \leq K_3\, e_{2,1} \leq (K_3 + 1)\, e_{2,1}$

For m > 2,

$$e_{m+1,m} \leq K_{m+1} \sum_{i=1}^{m-1} e_{mi}$$

$$\leq K_{m+1} \left(e_{m,m-1} + \sum_{i=1}^{m-2} \prod_{j=i+2}^{m} (K_j + 1) e_{i+1,i} \right)$$

$$\leq L_{m+1}(e_{m,n-1} + \sum_{i=1}^{m-2} \prod_{j=i+2}^{m} L_j \; e_{i+1,i})$$

$$+ \sum_{i=2}^{m-2} \left(\prod_{j=i+2}^{m} L_j \right) \left(2^{i-2} \prod_{j=3}^{i+1} L_j \right) e_{2,1}$$

$$\leq L_{m+1} \left[2^{m-3} + 1 + \sum_{i=2}^{m-2} 2^{i-2} \right] \prod_{i=3}^{m} L_j \; e_{2,1}$$

$$< \left[2^{m-3} + 1 + (2^{m-3} - 1) \right] e_{2,1} \prod_{i=3}^{m+1} L_i$$

$$< 2^{m-2} \; e_{2,1} \prod_{i=3}^{m+1} (K_i + 1).$$

This completes the proof.

Combining (14) and (18) completes the error analysis for the regular Gram-Schmidt process:

$$(18) \qquad e_{nm} \leq K_n \, e_{m+1,m} \left(\prod_{i=m+2}^{n-1} (K_i + 1) \right)$$

$$< \begin{cases} e_{2,1} \prod_{i=3}^{n} (K_i + 1) & \text{for } m = 1 \\ 2^{m-2} \; e_{2,1} \prod_{i=3}^{n} (K_i + 1) & \text{for } m > 2. \end{cases}$$

It is clear that e_{nm} will be maximized for a given n when m=n-1. This is also verified by experiment. Table 1 presents ε_{nm} for $k_{ni} = -9$.

It also appears that the error bounds for ε_{nm} are nearly achieved by the actual ε_{nm}. The above proof can be used to prove equality when the e_{nm} are replaced by the ε_{nm} if the approximation $K_i = K_{i-1} + 1$ is allowed. From the

closeness of the results in Table 1 it appears that is not too bad an approximation when $K_i = 9$. This result bears the same relationship to (18) as (16) does to (15) for the Modified Gram-Schmidt procedure.

The superiority of the Modified Gram-Schmidt procedure over the regular process is also verified by comparing Tables 1 and 2.

6. Method of the Linear Corrector. Formula (5) is a, very good approximate representation of the error for the regular Gram-Schmidt process. But, if we know in advance what the error is going to be, we should be able to eliminate it. For this purpose we formulate the regular Gram-Schmidt process with linear correctors d_{ni}. We shall then use (5) to determine optimum values for d_{ni}.

$$V_n = \alpha_n W_n - \sum_{i=1}^{n-1} (k_{ni} + d_{ni}) V_i. \tag{19}$$

Once again, ignoring errors of normalization and allowing $m > n$ for ϵ_{nm} produces an error formula:

$$\epsilon_{nm} = (V_n, V_m) = - \sum_{\substack{i=1 \\ i \neq m}}^{n-1} (kni + d_{ni}) (V_i, V_m) - d_{nm}$$

$$= - \sum_{\substack{i=1 \\ \text{ifm}}}^{n-1} (k_{ni} + d_{ni}) \epsilon_{im} - d_{nm}.$$

To determine $d_{n1}, d_{n2}, \ldots, d_{n,n-1}$ so that $\epsilon_{n1} = \epsilon_{n2} \cdots \epsilon_{n,n-1} = 0$ would require the solution of a system of n-1 linear equations in n-1 unknowns at each step of the Gram-Schmidt process. This is not a practical method. If,

on the other hand, we assume that this method will eliminate instability, we may take $|d_{ni}| << |k_{ni}|$, and then consider the system,

$$\epsilon_{nm} = - \sum_{\substack{i=1 \\ i \neq m}}^{n-1} k_{ni}\epsilon_{in} - d_{nm}.$$

Setting $\epsilon_{nm} = 0$ gives,

$$d_{nm} = - \sum_{i=1}^{m-1} k_{ni}\epsilon_{mi} - \sum_{i=m+1}^{n-1} k_{ni}\epsilon_{im}. \tag{20}$$

This assumption gives an efficient form for d_{nm}. It remains for experimental results to show if the assumption is valid. From computational results like Table 3, it appears to be so. The only appreciable additional computations are for the ϵ_{nm}. If these are being calculated anyway, then the method is a definite saving. However, it is less clear that the calculation of ϵ_{nm} purely for the use of this method is efficient.

Observe also that if there is no round-off error then all $d_{ni} = 0$ and we have the regular Gram-Schmidt process. Even when this condition does not hold the linear corrector will not take V_n 'out of the plane' since d_{ni} is a scalar like k_{ni}.

The exposition of the method of the linear corrector is in terms of k_{ni}. But k_{ni} cannot be used in a computational algorithm because it involves α_n, which is 'not available until the normalization of U_n is performed. However, the method of the linear corrector can easily be formulated in a manner which is applicable for a computational algorithm. The basic definition of the Gram-Schmidt process with linear corrector is,

$$U_n = W_n - \sum_{i=1}^{n-1} [(W_n,V_i) + D_{ni}]V_i$$

$$V_n = U_n/\| U_n \|.$$

Since $k_{ni} = \alpha_n (W_n,V_i)$, we get $D_{ni} = d_{ni}/\alpha_n$. From this and (20) we get,

$$\begin{aligned} D_{ni} &= d_{ni}/\alpha_n \\ &= \Big(- \sum_{\substack{i=1 \\ i\neq m}}^{n-1} k_{ni}\, \epsilon_{ni} \Big) / \alpha_n \\ &= - \sum_{\substack{i=1 \\ i\neq m}}^{n-1} (k_{ni}/\alpha_n)\epsilon_{ni} \\ &= - \sum_{\substack{i=1 \\ i\neq m}}^{n-1} (W_n,V_i)\epsilon_{ni}. \end{aligned}$$

This formula can easily be made into a computational algorithm.

Since the Modified Gram-Schmidt process is more efficient than the regular method, the possibility of using a linear corrector with the modified method arises. Proceeding as before:

$$V_n = \alpha_n^* W_n - \sum_{i=1}^{n-1} (k_{ni}^* + d_{ni}^*)V_i. \tag{21}$$

An error analysis like (9) gives,

$$\epsilon_{nm} = - \sum_{i=m+1}^{n-1} (k^*_{ni} + d^*_{ni})\, \epsilon_{im} - d^*_{nm}.$$

If we proceed as we did for (20) we get,

$$d^*_{nm} = - \sum_{i=m+1}^{n-1} k^*_{ni}\, \epsilon_{im}.$$

This is a simple form, except that k^*_{ni} cannot be evaluated until after d^*_{nm} has been determined (since $i \geq m+1$). However (10) gave us the result,

$$k^*_{ni} = k_{ni} + \theta(K_{ij}\epsilon_{ij}).$$

Since we have already assumed stability when we required $|d_{nm}| << |k_{nm}|$, we can once again assume $|\epsilon_{ij}| \leq .01$ and define,

(22) $$d^*_{nm} = - \sum_{i=m+1}^{n-1} k_{ni}\epsilon_{im}.$$

However, this requires that we evaluate k_{ni} as well as k^*_{ni} and ϵ_{nm}. It is doubtful that it represents a saving.

7. Iteration. Often none of the above methods will produce sufficiently accurate results. This often happens when α_n and k_{ni} are large. This is the result of the (ξ_n, V_m) term, which cannot be eliminated. In this case, the best procedure is to repeat the Gram-Schmidt process using the inaccurate V_n as the new W_n. Let $\bar{V}_n$ be the result of this second pass. Then, for the regular Gram-Schmidt method:

$$\bar{U}_n = V_n - \sum_{i=1}^{n-1} (V_n, \bar{V}_i)\bar{V}_i.$$

$$\bar{V}_n = \bar{U}_n / \|\bar{U}_n\|.$$

If the error in the V_n is'not too great then $\bar{V}_n$ will approximate V_n and $\bar{k}_{nm} \doteq \epsilon_{nm}$. Since the error in the $\bar{V}_n$ vectors is proportional to the $\bar{k}_{nm}$, which is of the same magnitude of the first pass errors, this iteration represents a 'second-order' method.

Iteration can be used with any modification of the Gram-Schmidt technique.

8. Numerical Experiments. In order to compare the three computation variations of the Gram-Schmidt process considered in this paper, and in particular to test (15) and (18), numerical experiments were conducted on an IBM 360/67 computer using long-precision arithmetic (14 hexadecimal digits). An example was constructed with $k_{ni} = -K_n = -9$ for n=1, 2, ..., 10. The results of this experiment are presented in Tables 1, 2, and 3. Table 1 presents ϵ_{nm} for the regular Gram-Schmidt process. Compare these results with (18). For $m \geq 2$ (18) becomes,

$$\epsilon_{nm} = 2^{m-2}\, e_{2,1} \prod_{i=3}^{n} (K_i + 1)$$

$$= 2^{m-2}\, 10^{n-1}\, \epsilon_{2,1}$$

since $(K_i + 1) = 10$.

This aspect of the formula is readily verified by examining any row or column of Table 1. It is apparent that the actual errors nearly obtain the

error bounds for e_{nm}. This suggests that the replacement of K_i by $K_i + 1$ does not adversely affect the accuracy of this heuristic error analysis. Observe that the largest error is $\epsilon_{10,9} = -2.9 \cdot 10^{-5}$. This element is the one expected to have the largest error. Had this calculation been done in regular precision arithmetic (6 hexidecimal digits) there would have been extreme instability after V_7.

Table 2 presents the ϵ_{nm} for the Modified Gram-Schmidt process applied to the same problem. Error formula (15) is verified here, as is clear from the columns of Table 2. As expected, the largest error is $\epsilon_{10,1} = -4.089 \cdot 10^{-7}$. Compare this value with the corresponding value for the regular Gram-Schmidt process, $\epsilon_{10,1} = -4.096 \cdot 10^{-7}$. The similarity demonstrates how the modified process prevents error propagation along the rows but not down the first column.

Table 3 presents ϵ_{nm} for the method of the linear corrector. The largest error is $\epsilon_{10,5} = 1.6 \cdot 10^{-14}$, which is all that could be expected from 14 digit-accurate computations.

Additional computations showed the method of the linear corrector to be comparable with the method of Householder transformations when both were applied to the Hilbert matrix of order 6.

We suggest that the basic utility of this paper is in presenting a method of orthonormalization which is comparable in accuracy with more sophisticated techniques and yet is both easy to understand and to program.

---

[1] John R. Rice, "Experiments on Gram-Schmidt Orthonormalization," Math. Comp., v. 20, 1966, pp. 325-328.

Table 1

-3.2000E-15
-4.1100E-14 -2.7807E-14
-4.0960E-13 -2.82393-I 3 -6.2449E-13
-4.0964E-12 -2.8247E-12 -6.2470E-12 -1.1843E-11
-4.0963E-11 -2.8244E-11 -6.2472E-11 -1.1844E-10 -2.2510E-10
-4.0962E-10 -2.8243E-10 -6.2472E-10 -1.1844E-09 -2.2510E-09 -4.2769E-09
-4.0962E-09 -2.8243E-09 -6.2472E-09 -1.1844E-08 -2.2510E-08 -4.2769E-08 -8.1261E-08
-4.0962E-08 -2.8243E-08 -6.2472E-08 -1.1844E-07 -2.2510E-07 -4.2769E-07 -8.1261E-07 -1.5440E-06
-4.09623-07 -2.8243E-07 -6.2472E-07 -1.1844E-06 -2.2510E-06 -4.2769X-06 -8.1261E-06 -1.5440E-05 -2.9335E-05

ϵ_{ij} for Regular Warn-Schmidt Process with $k_{n1} = -9$.

Table 2

-3.2000E-15									
-4.1200E-14	5.1106E-15								
-4.0900E-13	4.7628E-14	3.8241E-15							
-4.0894E-12	4.7133E-13	4.1301E-14	1.9582E-15						
-4.0894E-11	4.7177E-12	4.1196E-13	1.4656E-14	8.6018E-15					
-4.0893E-10	4.7179E-11	4.1183E-12	1.4688E-13	8.4567E-14	3.4853E-15				
-4.0893E-09	4.7179E-10	4.1188E-11	1.4721E-12	8.4267E-13	2.7187E-14	1.4537E-15			
-4.0893E-08	4.7179E-09	4.1187E-10	1.4726E-11	8.4282E-12	2.7896E-13	1.4204E-14	6.3004-E-16		
-4.0893E-07	4.7179E-08	4.1187E-09	1.4726E-10	8.4286E-11	2.7851E-12	1.3676E-13	1.9133E-15	3.3014E-15	

ϵ_{ij} for Modified Cram Schmidt Process with $k_{n1} = -9$.

Table 3

-3.2000E-15
-1.3200E-14 6.1939E-16
-1.1500E-14 -1.4794E-15 3.5223E-16
-1.0200E-14 -2.2280E-15 1.9980E-15 -3.8554E-16
-9.6000E-15 -4.6356E-16 6.6069E-15 1.4187E-16 1.1436E-14
-3.8000E-15 3.5273E-15 3.9285%15 3.9224E-15 1.0684E-14 6.3989E-15
-2.7000E-15 3.9701E-15 4.3554E-15 7.2906E-15 6.2020E-15 8.9330E-16 -2.1698E-15
-5.0000E-15 2.2657E-15 6.0209E-15 1.3287E-14 1.3287E-14 2.1319E-15 -6.8681E-15 7.4650E-15
-6.3000E-15 -8.4736E-16 4.7397E-15 1.5569E-15 1.5569E-14 3.0827E-15 -7.5095E-15 6.0888E-15 -2.4260E-15

ϵ_{ij} for Regular Gram-Schmidt **Process with** Linear Corrector, k_{ni} = -9.