The present paper contains in Table 1 a somewhat shortened version of that summary, and lists all values of $|k|$ for which six or more solutions were found.

We summarize some of our results as follows:

(1) For positive $k \leq 100$, no solution could be appended to the Table in [3].

(2) For negative $k \geq -9999$, the last solution found was

$$(1{,}775{,}104)^3 - (2{,}365{,}024{,}826)^2 = -5412;$$

whilst, for positive $k \leq 9999$, the last solution was

$$(939{,}787)^3 - (911{,}054{,}064)^2 = 307.$$

(3) In addition to solutions for $|k| \leq 9999$, we have solutions, for $y \geq 10^4$ and $|k| \leq 99999$; there are 1221 for positive k and 799 for negative k.

The vast majority of solutions are with $y < 100$ and Table 2 gives the number of solutions for various ranges of y.

The fact that the number of solutions is a rapidly decreasing function of y suggests that for at least some k the solution set may be complete.

Memorial University of Newfoundland
St. John's, Newfoundland, Canada

1. L. J. MORDELL, *A Chapter in the Theory of Numbers*, Cambridge at the University Press and Macmillan, New York, 1947. MR 8, 502.

2. B. M. DELONE & D. K. FADDEEV, *The Theory of Irrationalities of the Third Degree*, (English translation), Transl. Math. Monographs, Vol. 10, Amer. Math. Soc., Providence, R.I., 1964. MR 28 #3955.

3. O. HEMER, "Notes on the Diophantine equations $y^2 - k = x^3$," *Ark. Mat.*, v. 3, 1954, pp. 67–77. MR 15, 776.

4. W. LJUNGGREN, "On the Diophantine equation $y^2 - k = x^3$," *Acta Arith.*, v. 8, 1963, pp. 451–463. MR 28 #2082.

Experiments on Gram-Schmidt Orthogonalization

By John R. Rice*

1. Orthogonalization Procedures. In this note we present a brief resumé of some experiments made on orthogonalization methods. We have a set $\{u_i \mid i = 1, 2, \cdots, n\}$ of m-vectors and wish to obtain an equivalent orthonormal set $\{v_i \mid i = 1, 2, \cdots, n\}$ of m-vectors. We consider the following methods:

(a) *Gram-Schmidt* (GS). $v_1 = u_1/\| u_1 \|$.

$$v_k' = u_k - \sum_{j=1}^{k-1} (v_j, u_k)v_j, \quad v_k = v_k'/\| v_k' \|; \qquad k = 2, \cdots, n.$$

(b) *Modified Gram-Schmidt* (MGS). $v_1 = u_1/\| u_1 \|$,

$$u_j^{(1)} = u_j - (u_j, v_1)v_1, \qquad j = 2, \cdots, n.$$

$$\left. \begin{array}{l} v_k = u_k^{(k-1)}/\|u_k^{(k-1)} \| \\[2mm] u_j^{(k)} = u_j^{(k-1)} - (u_j^{(k-1)}, v_k)v_k \quad j = k + 1, \cdots, n \end{array} \right\} \quad k = 2, \cdots, n.$$

(c) *Modified Gram-Schmidt with pivoting* (MGS-Pivot). We assume that the vectors u_i are normalized so that $\| u_i \| = 1$, $i = 1, 2, \cdots, n$. Then, at the kth step $u_j^{(k-1)}$ is chosen so that

$$\| u_j^{(k-1)} \| = \max_{k \leq i \leq n} \| u_i^{(k-1)} \|.$$

Then $v_k = u_j^{(k-1)}/\| u_j^{(k-1)} \|$. The selection of $u_j^{(k-1)}$ may be made efficiently by noting that

$$\| u_i^{(k-1)} \|^2 = 1 - \sum_{l=1}^{k-1} (u_i^{(l)}, v_l)^2$$

and accumulating the required sum for each vector.

(d) *Reinforcement*. This is the repeated application of any of the first three methods.

This study was undertaken to investigate possible advantages of the pivoting. These turn out to be minor, but surprisingly, the Gram-Schmidt and modified Gram-Schmidt show distinct differences in computational behavior. This is particularly remarkable since both methods perform basically the same operations, only in a different sequence. Indeed, ignoring computational errors, they produce the same set $\{v_i\}$ with the same number of operations. The MGS method is more natural for machine computations since it economizes storage.

2. Experiments Performed. Experiments were made with 3 classes of vectors: (i) randomly generated, (ii) function generated and (iii) artificially contrived. The classes (i) and (iii) are self-explanatory, for class (ii) the vector $u_i = (u_{i1}, u_{i2}, \cdots, u_{im})$ is generated by $u_{ik} = f(i, k)$ where $f(\alpha, x)$ is some common mathematical function (e.g. $u_{ik} = \tan ik\, \theta$, $\theta = \pi/2m$, $k = 1, 2, \cdots, m$, $i = 1, 2, \cdots, n$). Sets of vectors from class (ii) are the most likely to occur in applications [1] and are also likely to offer computational difficulties.

The methods of orthogonalization were compared on two points: orthogonality of the vectors $\{v_k\}$ and invariance of subspace defined by $\{u_k\}$. In order to compare the subspaces spanned by $\{u_k\}$ and $\{v_k\}$, a double precision orthogonalization of the $\{u_k\}$ was performed. Then the angle between each computed $\{v_k\}$ and the subspace spanned by $\{u_k\}$ may be accurately computed. Care was taken that the double precision orthogonalization was, in fact, satisfactory and that the vectors obtained did, in fact, span the same subspace as $\{u_i\}$.

All inner products were made with double precision accumulative multiplication. The experiments were performed on an IBM 7094 with standard programming systems.

3. A Simple Error Analysis. We outline a simple error analysis of the GS method which supports the experimental conclusions reported in the next section. Assume that, for some value k, we have $(v_i, v_j) = \epsilon_{ij}$ for $i, j \leq k - 1$. Let

$$u_k = \sum_{i=1}^{k-1} \beta_i v_i + \eta$$

where $(\eta, v_i) = 0$, $1 \leq i \leq k - 1$. Then

$$v_k' = \eta - \sum_{j=1}^{k-1} \left(\sum_{i \neq j} \beta_i \epsilon_{ij} \right) v_j$$

and

$$\epsilon_{k,l} = \left[-\sum_{j=1}^{k} \left(\sum_{i\neq j} \beta_i \, \epsilon_{ij}\right) \epsilon_{jl} \right] \bigg/ \; \| v_k' \|.$$

Note that $\epsilon_{ll} = 1$ and we conclude that *nonorthogonality effects are magnified by* $1/\| v_k' \|$ *in the GS method.*

If, as is usual for class (ii) vectors, $\| \eta \|$ becomes small then $\| v_k' \|$ becomes small. This, in turn, increases the ϵ_{ij} causing v_k' to quickly become a linearly combination of the $\{v_i \mid i = 1, 2, \cdots, k - 1\}$. That is to say that $\epsilon_{k,k-1}$ quickly approaches 1.

On the other hand, the MGS method always has $\epsilon_{k,k-1} = 0$ within machine accuracy. Indeed

$$(u_j^{(k)}, v_k) = (u_j^{(k-1)}, v_k) - (u_j^{(k-1)}, v_k)(v_k, v_k) = 0.$$

4. Conclusions. It is not feasible to report in detail an analysis of the computational experiments. Rather, we summarize the experiments by a series of (experimental) conclusions and remarks.

(A) Randomly generated sets are unlikely to cause difficulty for any of the methods. Eighty sets of vectors ranging from $n = 5$, $m = 5$ to $n = 40$, $m = 80$ were tested. Only one set (for $n = 40$, $m = 40$) gave any method any difficulty. For this one case the GS method produced a set $\{v_k\}$ with maximum inner product of .00022413 compared with .0000 0568 and .0000 0522 for the MGS and MGS-Pivot methods, respectively.

(B) Function generated vectors tend to be linearly dependent. Twenty five sets $\{u_k\}$ were generated by the functions $P_3(x)x^\alpha$, $P_3(x)e^{\alpha x}$, $P_3(x) \sin \alpha x$, $P_3(x) \cos \alpha x$, $P_3(x) \tan \alpha x$, $P_3(x) \log (1 + \alpha x)$, rational functions and several combinations of these functions. $P_3(x)$ is a generic symbol for a polynomial of degree 3. Most experiments were for $n \sim 20$, $m \sim 20$ or 50.

(C) For this class, (ii), the MGS method is consistently, and often substantially, "stronger" than GS. That is to say that MGS obtains more orthogonal vectors than GS, and, in fact, one reinforcement of MGS was always sufficient to obtain an orthonormal set $\{v_k\}$ whereas GS frequently required 3 or 4 reinforcements and sometimes as many as 6 to obtain an orthonormal set.

(D) The extreme linear dependence in class (ii) results in poor subspace definition. This is reflected in the sets $\{v_k\}$ obtained after reinforcement and the maximum angle between a v_k and the space spanned by $\{u_k\}$ was usually of the order of 20°–40°. The MGS was usually, but not always, better than GS with regard to keeping the subspace invariant.

One exception occurs for the first (but not other) application of GS where the subspace always remained well defined. The simple error analysis of the GS method shows why this is to be expected.

(E) The experiments indicate that pivoting results in a perceptible, but small (even negligible), improvement, on the average, for classes (i) and (ii).

(F) For certain artificially contrived systems the MGS-Pivot method resulted in a significant improvement. One such example is $u_1 = (1, 0, \cdots, 0)$, $u_2 = (1, t, t^2, \cdots, t^{m-1})$, $u_3 = (0, 1, 0, \cdots, 0)$, $\cdots$, $u_m = (0, \cdots, 0, 1, 0)$ for $m \geq 10$ and $0 < t < \frac{3}{4}$.

(G) A basic difference in performance between GS and MGS is that once GS loses orthogonality, it produces almost identical vectors thereafter. The maximum inner product is usually .999 etc. (and this is to be expected from the error analysis of 3). On the other hand, MGS continues to generate distinct, if not orthogonal, vectors. Thus the inner products obtained (after orthogonality is lost) normally range between 0.1 and 0.9. This is no doubt due to the fact that v_k is always orthogonal to v_{k-1} within machine accuracy. This is suggested by the error analysis and confirmed by experiment.

I wish to acknowledge the able assistance of Rex Wolf in preparing the programs for these experiments.

1. P. J. DAVIS & P. RABINOWITZ, "Advances in orthonormalizing computation," *Advances in Computers*, F. L. Alt (Ed.), Vol. 2, Academic Press, New York, 1961, pp. 55–133. MR **25** # 1636.

A Matrix Reduction Problem

By J. W. Moon and L. Moser

1. Introduction. Let A_n denote an n by n matrix of 0's and 1's that is nonsingular over the field of residues modulo 2. Fine and Niven [1] have shown that these are $c_n 2^{n^2}$ such matrices where

$$c_n = \prod_{k=1}^{n} \left(1 - \left(\frac{1}{2}\right)^k\right).$$

Let $f(A_n)$ denote the minimum number of operations needed to transform A_n into the identity matrix I_n. (It may be necessary, of course, to interchange certain rows but we do not count this as an operation.) The object in this note is to give bounds for $f(A_n)$ which at least determine its order of magnitude for almost all matrices A_n. These may be of some interest in connection with the question of the minimal number of operations required to invert a matrix. Indeed our methods and results apply with only minor modifications to the case of matrices with real elements provided that in performing arithmetic operations only a fixed number of significant digits is retained.

THEOREM. *There exist positive constants c_1 and c_2 such that*

$$\frac{c_1 n^2}{\log n} < f(A_n) < \frac{c_2 n^2}{\log n}$$

for almost all matrices A_n; i.e. for all but a fraction which tends to zero as n tends to infinity.

2. A Lower Bound for $f(A_n)$. We will show that

$$f(A_n) > \lambda = \frac{(\frac{1}{2} - \epsilon) n^2}{\log_2 n}$$

for almost all matrices A_n where ϵ is an arbitrary positive constant.