Errors and Pertubations in Vandermonde Systems

by

J. M. Varah

Technical Report 90-24
July 17, 1990

Department of Computer Science
The University of British Columbia
Vancouver, British Columbia V6T 1W5
CANADA

Errors and Pertubations in Vandermonde Systems

J. M. Varah [*]

July 17, 1990

Abstract

The Björck-Pereyra algorithm for Vandermonde systems is known to produce extremely accurate results in some cases, even when the matrix is very ill-conditioned. Recently, Higham has produced an error analysis of the algorithm which identifies when this behaviour will take place. In this paper, we observe that this analysis also predicts the error behaviour very well in general, and illustrate this with a series of extensive numerical tests. Moreover, we relate the computational error to that caused by perturbations in the matrix elements, and show that they are not always commensurate. We also discuss the relationship between these error and perturbation estimates with the "effective well-condition" of Chan and Foulser.

1 Introduction

The numerical solution of Vandermonde systems of equations, in primal form $Vx = b$ or dual form $V^T y = c$, where

$$V = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & & \alpha_n \\ \vdots & \vdots & & \vdots \\ \alpha_1^{n-1} & \alpha_2^{n-1} & \dots & \alpha_n^{n-1} \end{bmatrix},$$

[*] Computer Science Department, University of British Columbia, Vancouver, Canada.

is of special interest, because the systems arise frequently and because they can be very ill-conditioned. The algorithms of choice for these systems were derived by Björck and Pereyra [2], and can be described as a UL factorization of V^{-1} (see Golub and Van Loan [5] pg. 178) with

$$L = L_{n-1} \dots L_1, \quad U = U_1 D_1 \dots U_{n-1} D_{n-1}, \tag{1}$$

where

$$L_k \equiv L_k(\alpha_k) = \left[\begin{array}{c|cccc} I_{k-1} & & & & \\ \hline & 1 & & & \\ & -\alpha_k & 1 & & \\ & & \ddots & \ddots & \\ & & & -\alpha_k & 1 \end{array}\right],$$

$$D_k^{-1} = \left[\begin{array}{c|ccc} I_k & & & \\ \hline & \alpha_{k+1} - \alpha_1 & & \\ & & \ddots & \\ & & & \alpha_n - \alpha_{n-k} \end{array}\right],$$

and $U_k = L_k^T(1)$. The primal system $Vx = b$ is solved via

$$x = ULb = (U_1 D_1 \dots U_{n-1} D_{n-1})(L_{n-1} \dots L_1)b \tag{2}$$

and the dual system $V^T y = c$ via $y = L^T U^T c$. We shall refer to both primal and dual algorithms as the BP algorithms. These BP algorithms are preferred over standard Gaussian elimination from the standpoint of computational effort since they require only $O(n^2)$ operations.

The accuracy of the BP algorithms has been a matter of some debate, with very high accuracy reported in some cases, even when the matrix V was ill-conditioned. Recently, in a series of papers, Higham [6], [7], [10] has shown that the error in the computed solution can be small (of the order of the machine precision) independent of the condition number of V, when the $\{\alpha_i\}_1^n$ are positive and increasing order.

In this paper, we attempt to clarify and extend Higham's excellent work, for the special case of monomial, non-confluent Vandermondes. In particular, we try to examine the error behaviour as the data vector changes, to relate the errors to changes in the solution caused by perturbations in the $\{\alpha_i\}$, and to estimate the error when other orderings of the $\{\alpha_i\}$ are used. We also relate the error behaviour to that of Gaussian elimination, and to the "effective well-conditioning" of Chan and Foulser [4], and compare the errors obtained in primal and dual systems. We also report on a series of extensive numerical tests.

2 Errors and Perturbations

In his first paper [6], Higham treats the BP algorithms as vector recurrences, and uses an ingenious argument based on majorizing sequences to produce a bound for the accumulated roundoff error in the computed solution $\hat{x}$ or $\hat{y}$, under the assumption that the α_i are positive and in increasing order. A simpler argument, based on the matrix formulation, is given in [10] and extended to systems arising from polynomials with three term recurrence relations.

Basically, the argument is as follows: for the primal system,

$$\hat{x} = f\ell(ULb) = \hat{U}\hat{L}b,$$

where

$$\hat{U} = \hat{U}_1\hat{D}_1 \ldots \hat{U}_{n-1}\hat{D}_{n-1},\ \hat{L} = \hat{L}_{n-1} \ldots \hat{L}_1$$

and each $\hat{L}_i = L_i + E_i, |E_i| \leq \eta|L_i|$ (and similarly for $\hat{U}_i, \hat{D}_i^{-1}$). Here η denotes the unit roundoff error. Thus

$$\hat{x} = x + Fb + O(\eta^2) \tag{3}$$

where F consists of all the first order terms in η in the difference $\hat{U}\hat{L} - UL$. One finds that

$$|F| \leq c(n,\eta)G$$

with

$$G = |U_1||D_1| \ldots |U_{n-1}||D_{n-1}|L_{n-1}| \ldots |L_1| \tag{4}$$

In [10], $c(n,\eta) = 8n\eta + O(\eta^2)$ but that analysis covers a more general case; in [6], the constant obtained in $5n\eta + O(\eta^2)$ and it is easily seen that this constant applies here as well if one is considering only Vandermonde systems.

So far, the analysis makes no restrictions on the $\{\alpha_i\}$. However, if $0 \leq \alpha_1 < \alpha_2 < \ldots < \alpha_n$, the diagonal elements of D_i are positive and U_i and L_i have checkerboard sign patterns, giving

$$|U_i| = DU_iD, \quad |L_i| = DL_iD, \quad |D_i| = DD_iD = D_i,$$

$D = diag(-1, +1, -1, -1, \ldots)$. This implies that

$$G = |U_1D_1 \ldots U_{n-1}D_{n-1}L_{n-1} \ldots L_1| = |U|\,|L| = |V^{-1}|$$

and hence that

$$|\hat{x} - x| \leq c(n, \eta)|V^{-1}|\,|b| \text{ and second order terms.}$$

Converting this to norms gives

$$\frac{\|\hat{x} - x\|_\infty}{\|x\|_\infty} \leq c(n, \eta)\theta(x)$$

where

$$\theta(x) \equiv \frac{\|\,|V^{-1}|\,|b|\,\|_\infty}{\|x\|_\infty}. \tag{5}$$

A similar bound holds for the dual system $V^Ty = c$.

As x (or b) varies, the error bound $\theta(x)$ varies significantly: in fact,

$$1 \leq \theta(x) \leq \kappa_s(V) \equiv \|\,|V^{-1}|\,|V|\,\|_\infty.$$

The maximum value ($\kappa_s(V)$ = Skeel condition number of V) is attained for $x = e = (1, 1, \ldots, 1)^T$ and the minimum value $\theta = 1$ is attained whenever $|b| = Db$ (that is, when the components of b alternate in sign).

Although $\theta(x)$ was derived as an error *bound*, it has been our experience that $\theta(x)$ also *estimates* the actual error incurred by BP algorithm very well. This is demonstrated in our numerical results in Section 5. Moreover, $\theta(x)$ can be efficiently estimated in practice using condition number estimators (see Higham [8]).

Clearly, $\theta(x)$ can be thought of as a measure of conditioning for the BP algorithms; the corresponding measure for Gaussian elimination without pivoting for Vandermonde systems is

$$\kappa_s(V, x) = \frac{\|\,|V^{-1}|\,|V|\,|x|\,\|_\infty}{\|x\|_\infty}.$$

We can use $\kappa_s(V,x)$ here in place of the usual (and larger) $\kappa(V)$ since V is totally nonnegative, and thus its computed LU factorization (without pivoting) has a small component-wise relative backward error, which implies a forward error involving $\kappa_s(V,x)$ not $\kappa(V)$. See Higham [10] and Arioli, Demmel, and Duff [1].

Since $\theta(x) \leq \kappa_s(V,x) \leq \kappa_s(V)$, we can expect the BP algorithm to be more accurate than Gaussian elimination in general, and again this is demonstrated by our numerical results (with a few notable exceptions).

In [10], Higham also shows that the computed solution $\hat{x}$ satisfies a backward error result — that is, $\hat{x}$ is the exact solution of a system $(V+E)\hat{x} = b$. However this new system is no longer Vandermonde, and the question still remains of whether the BP algorithm produces a computed solution $\hat{x}$ which is the exact solution of a nearby Vandermonde system - if so, then the BP algorithm would be strongly stable in the sense of Bunch [3].

In one sense this question is easily answered in the affirmative, as was pointed out by Higham [6], if one allows perturbations in the data vector b. Changes of order ε in b lead to changes in x of order $\theta(x) \cdot \varepsilon$ (whether primal or dual), and hence the computational error is equivalent to the error caused by perturbations in b of no more than $5n\eta$. Notice that this is the same kind of relationship as holds for perturbations and error in general linear systems when Gaussian elimination is used.

It is still of interest to consider as well the effect of perturbations in the matrix elements. Take the perturbed primal Vandermonde system

$$\overline{V}\overline{x} = b$$

with $\overline{V}$ having coefficients $\overline{\alpha_i} = \alpha_i(1+\varepsilon_i)$. Then, as shown in Higham [6], the corresponding change $(\frac{\|\overline{x}-x\|_\infty}{\|x\|_\infty})$ has as an attainable bound for its first order term in ε,

$$\varphi_P(x) = \frac{\| \, |V^{-1}HV| \, |x| \, \|_\infty}{\|x\|_\infty}. \tag{6}$$

Here H = diag (0, 1, ..., n-1).

Now the relationship between perturbation and error is not nearly as simple: although the functions $\varphi_P(x)$ and $\theta(x)$ are often comparable, they are not in all cases. In particular, since $\varphi_P(x)$ is maximized for $x = e$,

$$|\varphi_P(x)| \leq \varphi_P^{max} \equiv \|V^{-1}HV\|_\infty, \tag{7}$$

and although

$$\varphi_P^{max} \leq (n-1)\kappa_s(V) = (n-1)\theta_{max} = (n-1)\theta(e),$$

φ_P^{max} can be much smaller than $\kappa_s(V)$ - see Example 2 of Section 5. Even when these bounds are comparable, $\varphi_P(x)$ can be much smaller than $\theta(x)$. Take the case of a unit vector $x = e^{(k)}$:

$$\varphi_P(x) = \max_j |(V^{-1}HV)_{jk}|, \quad \theta(x) = \max_j (|V^{-1}|\,|V|)_{jk}.$$

In Example 1 of Section 5, $\varphi_p(x) << \theta(x)$ for several unit vectors $x = e^{(k)}$.

For the dual problem, the corresponding first order perturbation term is

$$\varphi_D(x) = \frac{\|\,|V^{-T}|\,|V^T Hx|\,\|_\infty}{\|x\|_\infty}. \tag{8}$$

This function is much more similar to $\theta(x)$ for the dual problem (aside from the degenerate case $x = e^{(1)}$); in fact, replacing H by I gives $\theta(x)$. We have not observed any significant differences in practice betwen $\theta(x)$ and $\varphi_D(x)$; for all our dual problems, the actual error was comparable to the perturbation caused by an $O(\eta)$ perturbation in the $\{\alpha_i\}$.

3 Relationship with Effective Well-Conditioning

In [4], Chan and Foulser point out for general linear systems $Ax = b$ that the sensitivity of the solution x to perturbations in b does depend on b, and in particular that even when A is very ill-conditioned, for those b primarily in the span of the smaller singular vectors of A, the sensitivity of x to changes in b is far less than what is predicted from the condition number of A.

In fact, if $Ax = b$ and $A(x + \delta x) = b + \delta b$, they show

$$\frac{\|\delta x\|_2}{\|x\|_2} \leq \gamma_k(x) \frac{\|\delta b\|_2}{\|b\|_2} \tag{9}$$

where

$$\gamma_k(x) = \frac{\sigma_{n-k+1}}{\sigma_n} \frac{\|b\|_2}{\|P_k b\|_2}.$$

Here $\{\sigma_i\}$ are the singular values of A in decreasing order, k is any value between 1 and n, and $P_k b$ denotes the projection of b into the span of the smallest k left singular vectors of A, $u^{(n-k+1)}, \ldots, u^{(n)}$.

Clearly there should be some connection between this result and the perturbation results used here for Vandermonde matrices - in particular between $\gamma_k(x)$ and $\theta(x)$, since $\theta(x)$ bounds the first order perturbation in x under perturbations in b. And indeed, for $b = u^{(n)}, \gamma_1(x) = \theta(x) = 1$. Similarly, for $b = u^{(n-k)}, \gamma_{k+1} = \frac{\sigma_{n-k}}{\sigma_n} \cong \theta(x)$. However, for other data vectors b, $\theta(x)$ gives a much better indication of "effective well-conditioning": for example $\theta(x) = 1$ for $b = e^{(j)}$ (i.e. when b is any unit vector). Yet all of the $\{\gamma_k(x)\}$ may be much larger - in Example 1 of Section 5, $\gamma_k(x) > 10^7$ for all k when $b = e^{(1)}$.

Note: These perturbation results involving $\theta(x)$ and $\gamma_k(x)$ hold for general systems, and hence a careful comparison of them is appropriate for general systems as well.

In addition to the perturbation result (9), Chan and Foulser also consider perturbations to the matrix elements which can lead to perturbations in the solution that are smaller than predicted by the usual condition number estimate. In particular, they show that a result similar to (9) holds when A is perturbed to $\overline{A}$ having a singular value decomposition close to A. They then conclude that an algorithm, whose error behaviour is equivalent to this kind of perturbation, can give results which exhibit this kind of reduced error.

Of course, this analysis is not needed for the BP algorithms because of the direct forward error result (5). However it is still of interest to ask whether nearby Vandermonde matrices $\overline{V}$ have singular value decompositions close to that of V. Unfortunately this is not the case, as can be easily seen in a 2 x 2 case. Consider

$$V = \begin{pmatrix} 1 & 1 \\ 1 & 1+\delta \end{pmatrix}$$

as our Vandermonde. This has singular values

$$\sigma_1 = 2 + \frac{\delta}{2} + \theta(\delta^2)$$

$$\sigma_2 = \frac{\delta}{2} + \theta(\delta^2).$$

Now take

$$\overline{V} = \begin{pmatrix} 1 & 1 \\ 1 & (1+\delta)(1+\varepsilon) \end{pmatrix}.$$

If $\overline{V}$'s singular value decomposition were close to V's, then

$$\left| \frac{\overline{\sigma_i} - \sigma_i}{\sigma_i} \right| = 0(\varepsilon) \text{ for } i = 1, 2.$$

However it is easy to see that

$$\overline{\sigma_2} = \frac{\delta + \varepsilon}{2} + 0(\delta^2) + 0(\varepsilon\delta).$$

Thus for δ small, we get a relative change in σ_2 of $0(\varepsilon/\delta)$ not $0(\varepsilon)$.

Hence the phenomenon of a perturbed matrix having a close singular value decomposition appears to be a fairly rare occurrence. One example is scaled diagonally dominant matrices; see Barlow and Demmel [1].

4 Different Orderings of the $\{\alpha_i\}$

If the $\{\alpha_i\}$ are not in increasing order, the differences $(\alpha_j - \alpha_i)$ for $j > i$ are not all of one sign, thus rendering the $\{D_k\}$ of (1) non-positive. This in turn means that the error matrix of (2.2),

$$G = |U_1|\,|D_1| \ldots |U_{n-1}|\,|D_{n-1}|\,|L_{n-1}| \ldots |L_1|$$

does not simplify to $|V^{-1}|$. In (4.8) of [10], Higham defines the ratio of the norms of these matrices as a measure of the extra sensitivity of the factorization.

It appears as though the matrix G could be substantially larger than $|V^{-1}|$ for an arbitrary ordering of the $\{\alpha_i\}$. However our numerical experience does not reflect this; typically the additional error with an arbitrary ordering is only 10-20%. We provide results for random ordering in our examples in the next section.

5 Numerical Results

In this section we give a fairly comprehensive set of results for some representative Vandermonde systems. The standard mode of computation was double precision on an Amdahl mainframe, with $\eta = 16^{-13} \cong 2.2 \times 10^{-16}$. To check the accuracy, double precision iterative refinement (IR) was used, iterating until convergence had taken place in all digits. This forced a limitation on the cases we could consider: the IR iteration had to converge. Although one would expect that this would limit the cases to those where the effective condition number was less than $1/\eta$, in practice IR converged for a much wider class of problems.

The first example is the original one used by Björck and Pereyra [2], with $\alpha_i = \frac{1}{n-i+3}, i = 1, \ldots n$. They produced an exact solution pair (x, b) which can be generalized as follows: define

$$p(x) \equiv (x-1)^{n+1} = \sum_{j=0}^{n+1} (-1)^j \begin{pmatrix} n+1 \\ j \end{pmatrix} x^j.$$

Since $p(1) = p'(1) = \ldots = p^{(n)}(1) = 0$, we have

$$\sum_{j=0}^{n+1} (-1)^j \begin{pmatrix} n+1 \\ j \end{pmatrix} j^k = 0 \text{ for } k = 0, 1, \ldots, n.$$

In particular, for arbitrary α and β,

$$\sum_{j=0}^{n+1} (-1)^j \begin{pmatrix} n+1 \\ j \end{pmatrix} (\alpha + \beta j)^k = 0, \quad k = 0, \ldots, n. \tag{10}$$

(Note: $(\alpha + \beta j)^k$ could in fact be any polynomial in j of degree n or less.) Now replace k by $(n-k)$ and move the first term to the other side:

$$\sum_{j=1}^{n+1} (-1)^{j-1} \begin{pmatrix} n+1 \\ j \end{pmatrix} (\alpha + \beta j)^{n-k} = \alpha^{n-k}, \quad k = 0, \ldots, n$$

or,

$$\sum_{j=1}^{n+1} \left[(-1)^{j-1} \begin{pmatrix} n+1 \\ j \end{pmatrix} (1 + \frac{\beta}{\alpha} j)^n \right] \frac{1}{(\alpha + \beta j)^k} = \alpha^{-k}, \; k = 0, \ldots, n.$$

This is the Vandermonde system $Vx = b$ with $\alpha_j = \frac{1}{\alpha + \beta j}, b_k = \alpha^{-k}$, and x_j in square brackets. Then taking $\alpha = 2, \beta = 1$, and rearranging the $\alpha's$ in increasing order gives the example. Others can be obtained using other values for α and β, and another set with integer $\alpha's$ by not replacing k by $(n-k)$ in (10):

$$\sum_{j=1}^{n+1}\left[(-1)^{j-1}\left(\begin{array}{c} n+1 \\ j \end{array}\right)\right](\alpha+\beta j)^{k}=\alpha^{k}, \quad k=0,\ldots,n.$$

We first present, in Tables 1 (primal) and 2 (dual), results for a wide variety of b's for n = 10. For each b, we give $\|x\|_\infty$, then the error $\frac{\|x-\hat{x}\|_\infty}{\|x\|_\infty}$ from the BP algorithm normalized by the limit roundoff η, first for increasing $\{\alpha_i\}$ and then for random $\{\alpha_i\}$, followed by the error estimate $\theta(x)$. Then in column 6 we give the same normalized perturbation in x caused by a perturbation in α of order η and in column 7 the corresponding estimate $\varphi(x)$. Finally in columns 8 and 9 we give the normalized Gaussian elimination error (without pivoting) and condition estimate $\kappa_s(x)$. For this example, results were much the same when partial pivoting was used, because the actual pivoting involved was trivial. The b's used are the singular vectors $\{u^{(k)}\}$ of V, the columns $\{a^{(k)}\}$, of V, and finally the vector used in [2] (and in Higham [6] as Example 1).

As can be seen from Tables 1 and 2, the BP algorithm is much preferred whenever b reflects the ill-condition of V, or equivalently when b has significant components in the direction of the smaller singular vectors of V. When this is not the case, then BP performs much like Gaussian elimination. There are even examples (e.g. $b=a^{(9)}$ in primal case) where the BP error is worse. Similar results occur for other values of n.

Finally, it is of interest to compare the performance of the primal and dual algorithms. Although the errors are larger in the dual case, this is due to the worse conditioning of the problem, not to any intrinsic shortcoming with the dual algorithm. For cases where $\theta(x)$ is close to 1, the dual algorithm produces the same excellent results as the primal algorithm. For the particular b of [2], where both [2] and [6] report worse errors in the dual case, the reason is clear: the vector b in the dual case is much more ill-conditioned. Indeed, the primal b has $\theta(x)$=650, the dual b has $\theta(x) > 10^{10}$. Moreover, the vectors b for the remaining primal examples of [6] are perfectly conditioned $(\theta(x)=1)$ for all n, whereas those for the dual problems get more badly conditioned as n increases.

Another reason for the apparent poor showing of the dual algorithm in [6] is the more stringent component-wise relative error measure used there:

b	$\|x\|$	BP err	BP(r) err	$\theta(x)$	pert	$\varphi(x)$	GE err	$\kappa_s(x)$
$u^{(1)}$	1	.76E9	.59E8	.54E10	.38E6	.84E6	.13E10	.54E10
$u^{(2)}$	2.8	.27E8	.27E9	.46E10	.18E6	.83E6	.29E10	.47E10
$u^{(3)}$	28	.63E9	.58E8	.35E10	.21E6	.83E6	.13E10	.50E10
$u^{(4)}$	540	.68E8	.52E8	.80E9	.30E6	.31E6	.11E10	.24E10
$u^{(5)}$	.88E4	.37E7	.21E8	.15E9	.10E6	.12E6	.30E9	.16E10
$u^{(6)}$	.24E6	.12E6	.59E6	.11E8	.71E4	.25E5	.87E8	.72E9
$u^{(7)}$	.86E7	.62E4	.52E5	.41E6	.14E3	.42E4	.42E8	.35E9
$u^{(8)}$	.39E9	.27E3	.15E3	.93E4	.16E3	.68E3	.15E8	.19E9
$u^{(9)}$	.29E11	29	27	.12E3	48	.13E3	.18E7	.11E9
$u^{(10)}$	.50E13	4.0	3.3	1.0	6.4	42	.30E6	.64E8
$a^{(1)}$	1.0	.14E6	.74E5	.62E7	45	49	.14E6	.62E7
$a^{(2)}$	1.0	.11E5	.15E6	.83E7	15	19	.44E5	.83E7
$a^{(3)}$	1.0	.33E5	.23F6	.12E8	7.9	9	.42E5	.12E8
$a^{(4)}$	1.0	.17E6	.16E6	.17E8	3.9	10	.12E7	.17E8
$a^{(5)}$	1.0	.16E6	.43E4	.28E8	10	15	.43E5	.28E8
$a^{(6)}$	1.0	.62E6	.26E7	.48E8	9.2	23	.26E7	.48E8
$a^{(7)}$	1.0	.20E7	.20E7	.98E8	55	77	.23E7	.98E8
$a^{(8)}$	1.0	.99E7	.13E7	.24E9	.41E3	.58E3	.85E7	.24E9
$a^{(9)}$	1.0	.70E8	.14E8	.79E9	.55E4	.11E5	.56E5	.79E9
$a^{(10)}$	1.0	.21E9	.31E9	.42E10	.76E5	.83E6	.45E9	.42E10
[2]	.91E8	51	6.1	650	8.2	46	.20E6	.67E8

Table 1: $\alpha_i = \frac{1}{(n-1+3)}$, $n = 10$, $primalcase$,
$\kappa = .13E14$, $\kappa_s = .55E10$, $\varphi_P^{max} = .84E6$

b	$\|x\|$	BP err	BP(r) err	$\theta(x)$	pert	$\varphi(x)$	GE err	$\kappa_s(x)$
$u^{(1)}$	.31	.12E9	.11E10	.13E14	.15E12	.23E12	.35E10	.13E14
$u^{(2)}$	3.5	.55E10	.60E11	.77E12	.59E12	.15E13	.25E11	.36E13
$u^{(3)}$	37	.11E10	.56E10	.49E11	.12E12	.36E12	.85E10	.14E13
$u^{(4)}$	.46E3	.24E9	.14E10	.46E10	.11E11	.41E11	.62E9	.56E12
$u^{(5)}$	.10E5	.87E7	.92E8	.30E7	.22E9	.22E10	.12E10	.15E12
$u^{(6)}$	.26E6	.97E6	.58E7	.12E8	.93E6	.11E9	.36E9	.38E11
$u^{(7)}$	.85E7	.12E5	.16E6	.39E6	.79E6	.50E7	.10E9	.10E11
$u^{(8)}$	.37E9	.13E3	.43E4	.95E4	.42E5	.14E6	.31E8	.25E10
$u^{(9)}$	.31E11	28	.11E3	.15E3	.95E3	.31E4	.75E7	.46E9
$u^{(10)}$	.58E13	3	3.6	1.0	14	38	.12E7	.63E8
$a^{(1)}$	1.0	0.0	0.0	12E14	0.0	0.0	0.0	.12E14
$a^{(2)}$	1.0	0.0	.31E-5	.13E13	.33E12	.13E13	.35E11	.13E13
$a^{(3)}$	1.0	.73E9	.16E12	.15E12	.91E11	.30E12	.64E10	.15E12
$a^{(4)}$	1.0	.23E8	.29E11	.17E11	.80E10	.51E11	.86E9	.17E11
$a^{(5)}$	1.0	.70E8	.23E10	.20E10	.37E9	.79E10	.57E8	.20E10
$a^{(6)}$	1.0	.32E7	.17E10	.24E9	.55E9	.12E10	.53E8	.24E9
$a^{(7)}$	1.0	.22E6	.33E9	.29E8	.36E8	.17E9	.62E7	.29E8
$a^{(8)}$	1.0	.14E6	.19E9	.37E7	.18E8	.26E8	.87E6	.37E7
$a^{(9)}$	1.0	.16E5	.83E8	.49E6	.28E7	.40E7	.16E6	.49E6
$a^{(10)}$	1.0	.63E4	.20E8	.69E5	.10E6	.62E6	.17E5	.69E5
[2]	580	.24E8	.13E9	.18E11	.27E10	.11E11	.10E10	.25E11

Table 2: $\alpha_i = \frac{i}{(n-i+3)}$, $n = 10, dual\ case$, $\kappa = .93E14$,
$\kappa_s = .14E14$, $\varphi_D^{max} = .17E13$

b	$\|x\|$	BP err	BP(r) err	$\theta(x)$	pert	$\varphi(x)$	GEPP	$\kappa_s(x)$
$u^{(1)}$	.12	.54E7	.34E8	.31E10	.23E4	.30E4	.47E8	.31E10
$u^{(2)}$	.31	.20E7	.59E8	.16E10	.83E3	.29E4	.54E8	.23E10
$u^{(3)}$	.97	.94E7	.14E8	.29E9	.17E4	.29E4	.13E7	.33E10
$u^{(4)}$	6.9	.65E5	.17E7	.61E8	.13E4	.16E4	.21E8	.24E10
$u^{(5)}$	53	.11E6	.39E6	.84E7	.22E3	.11E4	.64E8	.22E10
$u^{(6)}$	.60E3	.64E4	.22E5	.73E6	.18E3	.64E3	.10E9	.19E10
$u^{(7)}$	.93E4	.13E4	.29E4	.53E5	.27E3	.35E3	.45E8	.16E10
$u^{(8)}$	.23E6	.16E3	19	.25E4	27	.17E3	.33E8	.12E10
$u^{(9)}$	.86E7	18	14	72	11	90	.26E8	.83E9
$u^{(10)}$	.75E9	4.0	16	1.0	17	42	.63E8	.50E9

Table 3: $\alpha_i = 0.9^{i-1}$, $n = 10$, *primal case*, $\kappa = .80E10$, $\kappa_s = .41E10$, $\varphi_P^{max} = .30E4$

$\max_i(|\hat{x}_i - x_i|/|x_i|)$ rather than $\|\hat{x} - x\|_\infty/\|x\|_\infty$. For Example 6.2 of [6], the solution components are the coefficients of a Chebyshev polynomial, for which alternate coefficients are zero. The relative errors in these components are thus very large, which explains why the errors reported in Table 6.2 of [6] are large even for small n.

Our second example is the symmetric Vandermonde with $\alpha_i = w^{i-1}$. In Table 3 we give results for $b = u^{(k)}$, the singular vectors of V, for n=10 and w=0.9.

Results were again similar for other n; here the error using Gaussian elimination with pivoting was a factor of 10 better than that without pivoting, and hence we have given the former results. Notice that because the $\{\alpha_i\}$ are decreasing, not increasing, V is not totally non-negative, and thus the remarks in Section 2 on accuracy for Gaussian elimination without pivoting do not apply.

Our last example is the integer matrix with $\alpha_i = i$, used in Higham [8] as an example where Gaussian elimination without pivoting produces better results than with pivoting. Notice that in this case the matrix has some elements much larger than one so that the larger singular values are also much larger than one. In Tables 4 and 5 we give the results for $n = 10$ using

b	$\|x\|$	BP err	BP(r) err	$\theta(x)$	pert	$\varphi(x)$	GE err	$\kappa_s(x)$
$u^{(1)}$	.85E-9	.78E6	.17E7	.19E8	.26E3	.33E3	.26E7	.19E8
$u^{(2)}$	.12E-6	.33E6	.38E5	.50E7	.10E3	.23E3	.10E7	.21E8
$u^{(3)}$	.85E-5	.15E6	.27E5	.11E7	7.5	.13E3	.24E6	.16E8
$u^{(4)}$	.336-3	.72E4	.20E5	.23E6	35	81	.26E5	.12E8
$u^{(5)}$	.75E-2	.71E3	.12E4	.47E5	15	54	.34E3	.81E7
$u^{(6)}$	.089	.18E3	.41E3	.98E4	16	50	.13E3	.54E7
$u^{(7)}$	.57	27	.17E3	.15E4	16	49	.65E3	.27E7
$u^{(8)}$	1.9	34	59	.39E3	39	60	34	.42E7
$u^{(9)}$	25	5.1	12	38	4.9	39	23	.43E7
$u^{(10)}$	.11E4	4.3	4.3	1.0	6.2	23	41	.35E7

Table 4: $\alpha_i = i$, $n = 10, primal\ case$, $\kappa = .28E13, \kappa_s = .33E8$, $\varphi_P^{max} = .33E3$

again $b = u^{(k)}$.

For this example, the behaviour of Gaussian elimination is much more interesting, and we therefore expand on the results. As we mentioned earlier, since the matrix is totally nonnegative, Gaussian elimination without pivoting produces nonnegative L and U factors, and hence a guaranteed small component-wise relative backward error $\omega = O(\eta)$. This can be translated into a forward error bound using the perturbation analysis of Skeel [11]:

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq \frac{\omega \kappa_s(V, x)}{1 - \omega \kappa_s(V)}. \tag{11}$$

Notice however in Tables 4 and 5, the actual error observed using Gaussian elimination without pivoting is much smaller than that predicted by (11), and indeed is comparable to the BP error in many (but not all) cases.

When Gaussian elimination with partial pivoting is used, there is an immediate row interchange because of the large α's, and the total nonnegativity of the matrix is destroyed. This means that a small relative backward error ω cannot be guaranteed. In Table 6, we give the observed errors in the primal case for GE, GEPP, and after one step of (single precision) iterative refinement. Skeel [12] and Higham [9] have shown that under certain conditions

b	$\|x\|$	BP err	BP(r) err	$\theta(x)$	pert	$\varphi(x)$	GE err	$\kappa_s(x)$
$u^{(1)}$	.91E-9	.26E9	.14E12	.59E11	.15E12	.53E12	.19E11	.59E11
$u^{(2)}$	.18E-6	.96E8	.36E10	.19E10	.20E10	.10E11	.32E9	.14E11
$u^{(3)}$	.13E-4	.39E6	.12E9	.62E8	.30E8	.24E9	.10E8	.36E10
$u^{(4)}$	.47E-3	.11E5	.13E8	.23E7	.21E7	.10E8	.71E4	.11E10
$u^{(5)}$	.79E-2	.16E3	.17E6	.13E6	.37E6	.83E6	.21E5	.43E9
$u^{(6)}$	.086	16	.39E4	.89E4	.61E4	.76E5	.33E5	.15E9
$u^{(7)}$	.44	45	.49E4	.13E4	.37E4	.12E5	.52E4	.60E8
$u^{(8)}$	2.0	30	82	.35E3	.34E3	.37E4	.13E4	.40E8
$u^{(9)}$	26	3.7	34	36	.18E3	.46E3	.10E4	.17E8
$u^{(10)}$	.13E4	4.7	3.2	1.0	7.8	19	58	.35E7

Table 5: $\alpha_i = i$, $n = 10$, *dual case*, $\kappa = .26E13, \kappa_s = .67E11$, $\varphi_D^{max} = .59E12$

GEPP with one step of single precision interative refinement will produce a result with a small backward error ω, and this is demonstrated in Table 6. We also give the error bound (7.4) of Higham [8]:

$$\frac{\|x - \hat{x}\|_\infty}{\|x\|_\infty} \leq H(\hat{x}) = \frac{\|\, |V^{-1}|\, (|r| + \gamma_{n+1}\, (|b| + |V|\, |\hat{x}|\,))\, \|_\infty}{\|\hat{x}\|_\infty},$$

applied to the GEPP result ($\hat{x}$) and the refined result ($\hat{x}_{IR}$).

Again, the errors have been normalized by dividing by η. One step of IR reduced the error in GEPP considerably, but notice that GE (with no pivoting) can be even better. Incidentally, applying one step of single precision IR to the GE result is detrimental: the results are comparable to GEPP + IR. Notice also that the dominant term in $H(\hat{x}_{IR})$ is the one involving $\kappa_s(V, x)$.

References

[1] J. Barlow and J. Demmel. Computing accurate eigensystems of scaled diagonally dominant matrices. *SIAM J. Num. Anal.*, to appear, 1990.

b	$\|x\|$	BP err	GE err	GEPP($\hat{x}$)	H($\hat{x}$)	GEPP + IR	H($\hat{x}_{IR}$)
$u^{(1)}$	.85E-9	.78E6	.26E7	.11E11	.12E11	.37E7	.43E9
$u^{(2)}$	.12E-6	.33E6	.10E7	.60E9	.38E10	.15E7	.29E9
$u^{(3)}$	.85E-5	.15E6	.24E6	.11E10	.21E10	.48E6	.19E9
$u^{(4)}$	.33E-3	.72E4	.26E5	.90E9	.51E10	.17E6	.13E9
$u^{(5)}$	.75E-2	.71E3	.34E3	.50E9	.22E10	.24E6	.91E8
$u^{(6)}$	.89E-1	.18E3	.13E3	.25E9	.10E10	.53E5	.60E8
$u^{(7)}$	.57	27	.65E3	.95E8	.22E9	.87E4	.30E8
$u^{(8)}$	.19E1	34	34	.11E9	.47E9	.59E6	.47E8
$u^{(9)}$	.25E2	5.1	23	.53E8	.81E9	.20E6	.47E8
$u^{(10)}$	.11E4	4.3	41	.59E7	.28E9	.28E5	.39E8

Table 6: $\alpha_i = i,\ n = 10, primal\ case,\ \kappa = .28E13, \kappa_s = .32E8$

[2] A. Bjorck and V. Pereyra. Solution of Vandermonde systems of equations. *Math. Comp.*, 24:893–903, 1970.

[3] J. Bunch. The weak and strong stability of algorithms in numerical linear algebra. *Lin. Alg. Appl.*, 88:49–66, 1987.

[4] T. Chan and D. Foulser. Effectively well-conditioned linear systems. *SIAM J. Sci. Stat. Comp.*, 9:963–969, 1988.

[5] G. Golub and C.Van Loan. *Matrix Computations (Second Editions).* Johns Hopkins University Press, Baltimore, MD, 1989.

[6] N. Higham. Error analysis of the Bjorck-Pereyra algorithms for solving Vandermonde systems. *Num. Math.*, 50:613–632, 1987.

[7] N. Higham. Fast solution of Vandermonde-like systems involving orthogonal polynomials. *IMA J. Num Anal.*, 8:473–486, 1988.

[8] N. Higham. How accurate is Gaussian elimination? *Proceedings of 13th Dundee Biennial Conference on Numerical Analysis*, 1989.

[9] N. Higham. *Iterative refinement enhances the stability of QR factorization methods for solving linear systems.* Numerical Analysis Report Number 182, Dept. of Math., University of Manchester, 1990.

[10] N. Higham. Stability analysis of algorithms for solving confluent Vandermonde-like system. *SIAM J. Math. Anal.*, 11:23–41, 1990.

[11] R.D. Skeel. Scaling for numerical stability in Gaussian elimination. *J. Assoc. Comp.*, 26:494–526, 1979.

[12] R.D. Skeel. Iterative refinement implies numerical stability for Gaussian elimination. *Math. Comp.*, 35:817–832, 1980.