Roundoff Error Differences Between the Cray-2 and Cray Y-MP

Neil McCown[1]

Report RND-91-009, December 1991

NAS Systems Development Branch
NAS Systems Division
NASA Ames Research Center
Mail Stop 258-6
Moffett Field, CA 94035-1000

---

1. Computer Sciences Corp., NASA Contract NAS 2-12961, Moffett Field, CA 94035

Roundoff Error Differences Between the Cray-2 and Cray Y-MP

Neil McCown

Introduction

The Numerical Aerodynamic Simulation (NAS) Facility at NASA Ames Research Center is a leading center for state of the art scientific supercomputing. The typical computational problems attacked at NAS include large scale Computational Fluid Dynamics (CFD), structural dynamics, and computational chemistry simulations. NAS currently has two High Speed Processors (HSP's): a Cray 2, with 256 megawords (MW) of memory, 4 processors, and a 4.1 nanosecond (ns) clock cycle; and a Cray Y-MP 8-128, with 128 MW memory, 8 processors, and a 6.0 ns clock cycle.

Although the the Cray-2 and Y-MP at NAS have significant architecture differences, their floating point architectures are quite similar, and might be expected to produce similar computational results. However, a significant disagreement in the error of the solution to a numerical simulation produced on each machine by a NAS user was recently discovered. An investigation into the extent of the problem was undertaken, part of which is described below. The remaining details may be found in [1]. Specific issues addressed below include the relationship between problem size and the severity of the disagreement and whether this disagreement might affect the procurement of future HSPs.

The simulation modelled the aeroelastic behavior of the Space Shuttle's Solid Rocket Boosters. The model uses a variant of Cholesky factorization to solve a large sparse positive definite band matrix. On each Cray, seven tests were conducted that utilized FORTRAN LINPACK routines to solve linear systems similar to those of the simulation. Of these tests, four provided information on the total error arising in the solution of a model PDE, and three concerned the roundoff error arising in the solution of a simple matrix relation. When the seven sets of output were compared to the known exact (analytical) solution, the logarithm of the error was found to grow linearly with the logarithm of the problem size. By extrapolating these trends, predictions were made regarding the severity of the uncovered dis-

crepancy in future supercomputers.

Procedure

Four of the test problems that provided data for these predictions solved Laplace's equation on the unit square:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \nabla^2 u = 0$$

where u = f(x,y) for 0 x 1 and 0 y 1 on the boundary of the region. Laplace's equation was chosen because it is the simplest PDE which produces a banded symmetric positive definite matrix, the exact solutions for numerous boundary conditions were readily available, and the simplicity of the problem allowed the floating point operations to be modelled analytically.

The model PDE was discretized using central differences [2]. This method uses the approximation:

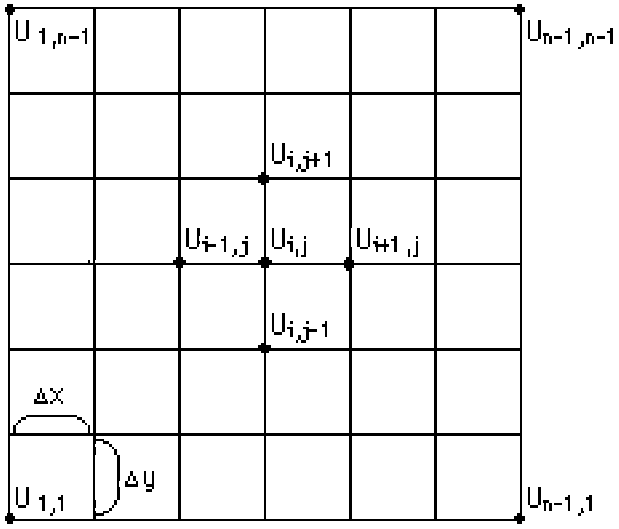
$$\nabla^2 u \cong \left[\delta_x^2 + \delta_y^2\right] U$$

where u is the exact solution and U is the solution to the discretized problem. From this relation, Laplace's equation at (i,j) becomes:

$$\left[\frac{\left[\frac{\Delta U_{left}}{\Delta x}\right] - \left[\frac{\Delta U_{right}}{\Delta x}\right]}{\Delta x}\right] + \left[\frac{\left[\frac{\Delta U_{lower}}{\Delta y}\right] - \left[\frac{\Delta U_{upper}}{\Delta y}\right]}{\Delta y}\right] = 0$$

where x and y represent the distance between nodes in the horizontal and vertical directions, respectively, and U represents the change in U in the specified direction.

Over the grid:



the grid spacing h is uniform, since $\Delta x = \Delta y = h$. Therefore,

$$\left[\frac{\left[\frac{U_{ij} - U_{i-1,j}}{h}\right] - \left[\frac{U_{i+1,j} - U_{ij}}{h}\right]}{h}\right] + \left[\frac{\left[\frac{U_{ij} - U_{ij-1}}{h}\right] - \left[\frac{U_{ij+1} - U_{ij}}{h}\right]}{h}\right] = 0$$

and, simplifying,

$$4U_{ij} - U_{i-1,j} - U_{i+1,j} - U_{ij-1} - U_{ij+1} = 0.$$

The entire set of algebraic equations is then solved simultaneously to determine the unknowns. Since the solutions at the grid boundaries are specified in the problem, equations are needed only for interior nodes.

The system of equations is then assembled into a coefficient matrix A. In describing the matrix A, N_X and N_Y are defined as the number of nodes in the coordinate directions and n the order of the system:

$$n = (N_X - 1)^2 = (N_y - 1)^2$$

The first element of each row corresponds to the U(1,1) term, the second to the U(1,2) term, the (N_X-1)th to the U(1,n-1) term, the nth to the U(2,1) term, and the last to U(n-1,n-1); n is the dimension of the matrix. Equations for the nodes closest to the boundaries have constants (for the known boundary conditions); these terms are collected in the right hand side vector b. The set of equations leads to the matrix relation:

$$
\begin{bmatrix}
-4 & 1 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & & \cdots & & 0 \\
1 & -4 & 1 & 0 & 0 & \cdots & 0 & 1 & 0 & & \cdots & & 0 \\
0 & 1 & -4 & 1 & 0 & \cdots & 0 & 0 & 1 & & \cdots & & 0 \\
\vdots & & \ddots & \ddots & \ddots & & & & & \ddots & & & \vdots \\
0 & 0 & \cdots & 1 & -4 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\
1 & 0 & \cdots & 0 & 0 & -4 & 1 & 0 & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \\
0 & 1 & \cdots & 0 & 0 & 1 & -4 & 1 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 \\
\vdots & & \ddots & & & & \ddots & \ddots & \ddots & & & & & \ddots & \vdots \\
0 & \cdots & & 1 & 0 & 0 & \cdots & 1 & -4 & 0 & 0 & 0 & 0 & \cdots & 1 \\
0 & \cdots & & 0 & 1 & 0 & \cdots & 0 & 0 & -4 & 1 & 0 & 0 & \cdots & 0 \\
0 & \cdots & & 0 & 0 & 1 & \cdots & 0 & 0 & 1 & -4 & 1 & 0 & \cdots & 0 \\
\vdots & & & & & & \ddots & & & & \ddots & \ddots & \ddots & & \vdots \\
0 & & \cdots & & & & 1 & 0 & 0 & \cdots & 1 & -4 & 1 & 0 \\
0 & & \cdots & & & & 0 & 1 & 0 & \cdots & 0 & 1 & -4 & 1 \\
0 & & \cdots & & & & 0 & 0 & 1 & \cdots & 0 & 0 & 1 & -4
\end{bmatrix}
\begin{bmatrix}
U_{1,1} \\ U_{2,1} \\ U_{3,1} \\ \vdots \\ U_{n-1,1} \\ U_{1,2} \\ U_{2,2} \\ \vdots \\ U_{n-1,n-2} \\ U_{1,n-1} \\ U_{2,n-1} \\ \vdots \\ U_{n-3,n-1} \\ U_{n-2,n-1} \\ U_{n-1,n-1}
\end{bmatrix}
=
\begin{bmatrix}
-b1_{1,0} - b4_{0,1} \\ -b1_{2,0} \\ -b1_{3,0} \\ \vdots \\ -b1_{n-1,0} - b2_{n,1} \\ -b4_{0,2} \\ 0 \\ \vdots \\ -b2_{n,n-1} \\ -b4_{0,n-1} - b3_{1,n} \\ -b3_{2,n} \\ \vdots \\ -b3_{n-3,n} \\ -b3_{n-2,n} \\ -b2_{n,n-1} - b3_{n-1,n}
\end{bmatrix}
$$

where b1, b2, b3, and b4 represent the boundary conditions at y=0, x=1, y=1, and x=0, respectively.

The boundary conditions and analytical (exact) solutions distinguished the four model problems, obtained from [3] and [4]. These were:

model problem 1, for $0 < x < 1$, $0 < y < 1$,

$$u(x,0) = \sin^2 \pi x,$$
$$u(x,1) = u(0,y) = u(1,y) = 0$$

exact solution:

$$u(x,y) = -(8/\pi)\sum_{1}^{\infty} \frac{1}{(2n-3)(4n^2-1)} \frac{\sinh(2n-1)(\pi-y)}{\sinh(2n-1)\pi} \sin(2n-1)x \; ;$$

model problem 2, for $0 < x < 1$, $0 < y < 1$,

$$u(x,0) = x^2\pi^3(1-x),$$
$$u(x,1) = u(0,1) = u(1,1) = 0$$

exact solution:

$$u(x,y) = -4\sum_{1}^{\infty} [1+2(-1)^n] \frac{\sinh n(\pi-y)}{n^3 \sinh n\pi} \sin nx \; ;$$

model problem 3, for $0 < x < 1$, $0 < y < 1$,

$$u(x,0) = u(x,1) = x^2\pi^2,$$
$$u(0,y) = 0,$$
$$u(1,y) = \pi^2$$

exact solution:

$$u(x,y) = \pi x - (8/\pi)\sum_{1}^{\infty} (2n-1)^{-3} \frac{\cosh(2n-1)(\frac{1}{2}\pi-y)}{\cosh(n-\frac{1}{2})\pi} \sin(2n-1)x \; ;$$

model problem 4, for $0 < x < 1$, $0 < y < 1$,

$$u(x,0) = u(x,1) = \begin{cases} 2x & \text{for } 0 < x \le 0.5, \\ 2-2x & \text{for } 0.5 < x < 1, \end{cases}$$

$$u(0,y) = u(1,y) = 0$$

exact solution:

$$u(x,y) = [8/\pi^2] \sum_{1}^{\infty} \frac{\sin \frac{n\pi}{2} [\sinh n\pi y + \sinh n\pi (1-y)]}{n^2 \sinh n\pi} \sin n\pi x \, .$$

Since the analytical solution to each model problem was an infinite series, a subroutine was written to find the sum using a DO loop.

Although Laplace's equation was a convenient PDE for examining error growth, two additional errors arose that obscured the effects of round-off error. The first was discretization error in the solution, a result of the approximate nature of the finite differences method. The second was roundoff error in the right hand side vector b, due to the fact that the machine rounded off of entries (boundary conditions) not exactly representable as binary floating point numbers. Three additional tests, suggested by L. Lustman of NAS, were therefore developed to avoid these errors. These three tests solved the matrix relation Ax=b, A being the matrix generated by the central differences method, were therefore developed to avoid these errors. These three tests solved the matrix relation Ax=b, A being the matrix generated by the central differences method, and x one of three pre-defined one-dimensional vectors:

case 1,

$$x_1 = [1, 1, \ldots \quad 1]^T$$

n = 1600, 6400, 14400;

case 2,

$$x_2 = \left[\frac{40000}{n}, \; 2 \times \frac{40000}{n}, \; 3 \times \frac{40000}{n}, \ldots i \times \frac{40000}{n}, \ldots (n-1) \times \frac{40000}{n}, \; 40000 \right]^T$$

n = 64, 400, 6400, 40000

case 3,

$$x_s = \left[\frac{99856}{n},\ 2 \times \frac{99856}{n},\ 3 \times \frac{99856}{n},\ \ldots\ i \times \frac{99856}{n},\ \ldots\ (n-1) \times \frac{99856}{n},\ 99856 \right]^T$$

n = 256, 1024, 16384, 99856

The machine being tested represented each x vector exactly, without rounding off any entries. Since all entries in the A matrix were integers, the product of the matrix multiplication was also represented exactly. Consequently, it had no roundoff error. The test code itself computed this right hand side vector, then passed it to the LINPACK matrix solver subroutines. By comparing the solution returned to the original vector x, machine roundoff error could be examined more accurately, without the errors of the finite differences method.

The LINPACK collection [5] provides a pair of subroutines, SGEFA and SGESL, which solve general square matrices such as the coefficient matrix. The fact that the coefficient matrix, which had as many elements as the square of the number of interior nodes, exceeded the Y-MP's memory limitation with a relatively large grid spacing limited the usefulness of these routines. The SPBFA and SPBSL subroutines alleviate this problem by taking advantage of the banded, symmetric nature of the coefficient matrix. Rather than operate on the entire matrix, they required a matrix with only those elements above the main diagonal and within the band. For example, consider a symmetric matrix of order seven and with two bands above the main diagonal in the form used by SGEFA and SGESL:

$$\begin{matrix}
11 & 12 & 13 & 0 & 0 & 0 & 0 \\
12 & 22 & 23 & 24 & 0 & 0 & 0 \\
13 & 23 & 33 & 34 & 35 & 0 & 0 \\
0 & 24 & 34 & 44 & 45 & 46 & 0 \\
0 & 0 & 35 & 45 & 55 & 56 & 57 \\
0 & 0 & 0 & 46 & 56 & 66 & 67 \\
0 & 0 & 0 & 0 & 57 & 67 & 77
\end{matrix}$$

If SPBFA and SPBSL solved the same matrix, however, the following form would be used:

```
 *   *  13  24  35  46  57
 *  12  23  34  45  56  67
11  22  33  44  55  66  77
```

where the *s denote elements that are not used.

Individual jobs were submitted to the NAS Network Queuing System (NQS). The Y-MP's upper NQS memory limit is 64 megawords, and since each element is a word, the programs were limited to 64 million array elements. Since the program required a one dimensional array with n elements for the exact solution and for the numerical solution, and a two dimensional array with elements for the band matrix, the total number of array elements was:

$$\text{elements} = n + n + (N_x\text{-}1)n = 3n + (N_x\text{-}2)n,$$

but since $n = (N_x - 2)^2$, this equation simplified to:

$$\text{elements} = (N_x - 2)^3 + 3(N_x - 2)^2$$

For 64 million elements, $(N_x\text{-}2)$ was found to be 399 and the maximum order of the linear system 15920 2. If SGEFA and SGESL were used instead of SPBFA and SPBSL, a two dimensional array with elements would be re-

$n \times n$

quired. The program would use a total of $(N_x\text{-}2)^4 - 2(N_x\text{-}2)^2$ elements, which would limit the order to 79999.

Finally, the solution vector returned by SPBSL was used to obtain the relative and absolute error. The maximum norm of the difference between this vector and the analytical solution is defined as the absolute error; the quotient of the absolute error and the maximum norm of the analytical solution is the relative error.
That is:

$$\text{absolute error} = \left\| \text{analytical solution} - \text{numerical solution} \right\|_\infty$$

$$\text{relative error} = \frac{\left| \text{absolute error} \right|}{\left\| \text{analytical solution} \right\|_\infty}$$

For the three tests that did not solve Laplace's equation, the x_1, x_2, and x_3 matrices replaced the analytical solution.

In addition to varying the problem's boundary conditions, solutions were obtained for a range of grid spacings. A UNICOS bourne shell script file was created that compiled and executed each program with a grid spacing of 1/10. Then, using a borne shell *do...done* loop and the *sed* command, the script decremented the spacing by 1/10 to the minimum allowable value. Since grid spacing was directly related to problem size, the maximum order of 159202 resulted in a minimum grid spacing of 1/400.

Each job, consisting of 40 runs, was broken down into several smaller jobs. As much data as possible was obtained using the daytime NQS queues, which allowed grid spacings down to 1/250. The job accounting reports indicated that the remaining solutions would require approximately 6000 seconds on the Y-MP, and 10000 seconds on the Cray-2. Since these jobs would be in relatively large queues, only two jobs could be submitted at a time. Those queues are not normally activated daily and generally have several other very large jobs waiting to execute. Model problem 4 was therefore the only problem executed with every remaining grid spacing, time constraints prevented the execution of problems 1, 2, and 3 with all grid spacings. For these problems, predictions were made from the day NQS jobs and from a night NQS job at the minimum grid spacing (1/400).

Results

The difference between the relative errors on the Y-MP and the Cray-2, e, was obtained for each run from the relation:

$$e = \text{relative error Y-MP} - \text{relative error Cray-2}$$

Although small compared to the relative error, e was increased as grid spacing decreased. Plot 1 is a graph of the logarithm of e versus the logarithm of the number of nodes for each of the tests conducted. Least squares model fits, with coresponding correlation coefficients in Table 1. With the exception of the Model 4 run, the data show differences in relative error that vary as a power of n with exponents larger than expected maximum exponent of one. Thus, it can be expected that for increasing n, a point will be reached at which the dominant error component for Y-MP computed solutions of the examined model problems will be due to the particular characteristics of Y-MP floating point arithmetic.

Test case	De model fit	Correlation Coefficient	De extrapolation to TFLOP system
Model 1	$10^{-16.1}\, n^{1.34}$	1.000	6.58×10^{-7}
Model 2	$10^{-15.3}\, n^{1.01}$	0.998	1.60×10^{-8}
Model 3	$10^{-15.7}\, n^{1.40}$	1.000	5.75×10^{-6}
Model 4	$10^{-15.1}\, n^{0.85}$	0.999	1.57×10^{-5}
Test 1	$10^{-15.4}\, n^{1.34}$	1.000	3.93×10^{-6}
Test 2	$10^{-17.3}\, n^{1.75}$	0.999	5.23×10^{-5}
Test 3	$10^{-15.4}\, n^{1.34}$	1.000	3.36×10^{-6}

Table 1.

Effect of Teraflop System Sized Problems on Accuracy

Although e was small in the data collected, the fact that it increased indicates that it could be substantial in larger machines. The capabilities of large scale computer systems are projected to increase significantly in the next ten years. Projections of the systems to be installed at NAS, provided by D. Pase at NAS, indicate that these systems will eventually exceed a Teraflop in computation rate. These systems would have in excess of 1.37×10^{11} words of main memory, which corresponds to:

$$1.37 \times 10^{11} \text{ words} = (N_x-2)^3 + 3(N_x-2)^2$$
$$(N_x-2) = 5159.65$$
$$(N_x-2)^2 = n = 2.66 \times 10^7$$

and using Test 2,

$$De = 10^{-17.3} (2.66 \times 10^7)^{1.75}$$
$$= 5.23 \times 10^{-5}$$

Thus for this size problem and the Test 2 right hand side, Y-MP floating point arithmetic is expected to produce a result that differs in the 5th decimal digit from the result expected from Cray 2 floating point arithmetic. Estimates for the other six test cases are listed in Table 1 in the column labeled "De extrapolation to TFLOP system."

Summary

The results of this investigation show that for the model problems examined, a significant correlation exists between the size of the problem and the amount the Y-MP and Cray-2 differ in relative error. While this conclusion may not hold for every application, banded symmetric matrices appear frequently in many common applications, and the results indicate that for these types of problems Cray 2 arithmetic is preferable, particularly as the problem size scales to the capacity of future supercomputers.

Acknowledgments

The author would like to thank Russell Carter, Duane Carbon, and David Browning for advice on making improvements to this report; Levi Lustman for suggesting additional tests for roundoff error differences; and Douglas Pase for his information regarding Teraflop machines.

References

[1] R. Carter (1990). "Y-MP Floating Point and Cholesky Factorization," RND-90-007, Numerical Aerodynamic Simulation Facility, NASA Ames Research Center.

[2] E. Kreyszig (1988). *Advanced Engineering Mathematics, Sixth Edition,* pp.1083-86.

[3] H. F. Weinberger (1965). *A First Course in Partial Differential Equations,* pp.97-99.

[4] D. L. Powers (1979). *Boundary Value Problems, Second Edition,* pp. 181-82.

[5] J. J. Dongarra, C. B. Moler, J. R. Bunch, G. W. Stewart (1984). *LINPACK Users' Guide,* pp. 1.1-1.4, 4.1-4.5.