NASA Technical Memorandum 82856

Film Shape Calculations on Supercomputers

Bernard J. Hamrock
Lewis Research Center
Cleveland, Ohio

Prepared for the
1982 Joint Lubrication Conference
cosponsored by the American Society of Mechanical Engineers
and the American Society of Lubrication Engineers
Washington, D.C., October 4-6, 1982

NASA

FILM SHAPE CALCULATIONS ON SUPERCOMPUTERS

Bernard J. Hamrock
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135

E-1225

Abstract

A brief description of supercomputers is presented along with a definition and illustration of vectorization. To demonstrate the usefulness of these supercomputers in solving tribological problems, a simple kernel of the film shape calculations in an elastohydrodynamic lubricated rectangular contact is presented. The relevant equations are briefly described. Both scalar and vector versions of the film shape code are presented. The run times of the two types of code indicate that over a 50-to-1 speedup of scalar to vector computational time for vector lengths typically used in elastohydrodynamic lubrication analysis is obtained.

Introduction

The present day powerful computing machines known as supercomputers have peak computing speeds exceeding 100 million operative per second. This is to be contrasted with the first commerical electronic computer delivered in 1951 which had a peak computing speed of around 600 operations per second. This means that the speed of large scale scientific computers has doubled on the average every 2 years. Although the current performance levels of such machines owe much to the rapid advance of microelectronics, new concepts in computer architecture have been equally important. The term "architecture" refers to the logical organization of the computer as it is seen by the programmer. The recent architectural innovations of greatest significance are those that enable the machine to carry out many similar operations in parallel, or concurrently. This is referred to as vector computation. The latest supercomputers allow the programmer to specify that many different elementary steps be executed simultaneously, whereas earlier computers obliged the programmer to break his computational program down into a sequence of elementary steps that would be executed one at a time. This is referred to as scalar computation. The speed with which signals can be propagated from one part of the computer to another is limited by the speed of light, which is close to 0.3 m per nanosecond. To get any further improvement in computation speed meant that innovative approaches to the basic architecture of the computer were required. These circumstances brought about the development of the latest supercomputers which utilize vector operations to achieve faster operational speeds. Levine (1982) gives a thorough and useful description of supercomputers.

The latest supercomputers represent a quantum jump in the speed of operation of large scientific computers, and their effect in solving tribological problems of the next decade should be significant. The present paper will attempt to demonstrate the effect of using these computers in performing film shape calculations in solving elastohydrodynamic lubrication problems. Comparisons will be made between scalar and vector operations to demonstrate the utility of new computers. The film shape calculations were chosen since in elastohydrodynamic lubrication calculations over 80 percent of the computation time is spent in the film shape calculations. For simplistic considerations a rectangular contract (sometimes referred to as a line contact) is studied rather than an elliptical contact.

Film Shape Formulation

Only the film shape calculations of the elastohydrodynamic lubrication analysis will be considered in this paper. The film shape for a rectangular contact can be written as:

$$h = h_o + S(x) + \delta(x) \tag{1}$$

where:

h_o = constant

$S(x)$ = separation due to geometry of undeformed solids

$\delta(x)$ = elastic deformation

The separation due to the geometry of the underformed rollers shown in figure 1(a) can be described by an equivalent cylindrical solid near a plane as shown in figure 1(b). The geometrical requirement is that the separation of the two rollers in the initial and equivalent situations should be the same at equal values of x. Therefore the separation due to the underformed geometry of the two rollers can be written as:

$$S(x) = \frac{x^2}{2R_x} \tag{2}$$

where:

$$\frac{1}{R_x} = \frac{1}{r_{ax}} + \frac{1}{r_{ay}} \tag{3}$$

Equation (2) makes use of the well-known parabolic approximation to the circular section of the solid and is valid as long as the separation is much smaller than the radius of curvature.

From Timoshenko and Goodier (1951), the elastic deformation at a point of a semi-infinite solid subjected to a pressure p along the line $x = x_1$, can be written as:

$$\delta(x) = -\frac{2}{\pi E'} \int_{-\bar{b}}^{\bar{b}} p \ln(x - x_1)^2 \, dx_1 \tag{4}$$

where:

$$E' = \frac{2}{\frac{1 - v_a^2}{E_a} + \frac{1 - v^2}{E_b}} \tag{5}$$

v = Poisson's ratio

E = modulus of elasticity

Figure 2 shows a rectangular area of uniform pressure. If the pressure is assumed to be uniform over the rectangular area the pressure can be put in front of the integral in equation (4). Therefore the integration can be performed, resulting in the following:

$$\delta(x) = \frac{2}{\pi} P \; D \tag{6}$$

where:

$$D = b\left[(X - B)\ln(x - B)^2 - (X + B)\ln(X + B)^2 + 4B(1 - \ln b)\right] \tag{7}$$

$$P = p/E' \tag{8}$$

b = semiwidth of Hertzian contact

X = x/b, dimensionless coordinate

$B = \bar{b}/b = \frac{1}{2n}$

n = number of nodes within the semiwidth of the contact

In equation (6) the elastic deformation is found to be equal to a constant multiplied by the dimensionless pressure multiplied by a distance influence factor. Now the term $\delta(x)$ in equation (6) represents the elastic deformation at a point A due to a rectangular area of uniform pressure and width 2 $\bar{b}$. If the conjunction is divided into a number of equal rectangular areas, the total deformation at a point A due to the contribution of the various rectangular areas of uniform pressure in the conjunction can be evaluated numerically. The total elastic deformation caused by the rectangular areas of uniform pressure within a conjunction can be written as:

$$\delta_k(x) = \frac{2}{\pi} \sum_{i=1,2,\ldots} P_i \; D_j \tag{9}$$

where:

$$j = |k - i| + 1 \tag{10}$$

Therefore substituting equations (2) and (9) into equation (1) while writing the film shape in dimensionless form gives:

$$H_k = \frac{h_k}{R_x} = H_0 + \frac{1}{R_x}\left[\frac{X^2}{2}\left(\frac{b^2}{R_x}\right) + \frac{2}{\pi} \sum_{i=1,2,\ldots} P_i\ D_j\right] \tag{11}$$

Scalar and Vector Programs

Having formulated the film shape, the next task was to program it both scalarly and vectorially. Before proceeding it might be well to point out more specifically what is meant by scalar and vector coding. Scalar coding is conventional coding that has been in existence since the first commerically available computer in 1951. Vectorization, or the process of writing programs in vector code, consists of designing, organizing, and writing programs so the maximum possible number of arithmetic and logical operations are executed as vector instructions. Vectorization is most easily introduced and illustrated by example. Consider arrays A and B, each consisting of 100 numbers. Assume that one wishes to compute array C, where $C_i = A_i + B_i$, $i = 1,\ldots,100$. The traditional "scalar" computer executes five assembly language instructions 100 times. There are two memory fetches (A_i and B_i), one addition, one store to memory (for C_i), and an instruction that increments a counter, tests, and branches back to load the next pair of input operands. Thus, 500 scalar instructions are executed to add arrays A and B. A vector computer's compiler can generate "vector" object code, which executes very differently. The vector code for adding the 100 pairs of operands consists of a single hardware instruction. The vector instruction executes by continuously streaming operands from central memory into the central processor, where the addition takes place, and continuously streaming answers back to memory. During execution of the vectorized addition, some elements of A and B are being read from central memory, some elements of A and B are undergoing addition, and some are being written in memory.

Figures 3 and 4 show the computer code for the film shape of the scalar and vector kernels, respectively. The only calculations given in these figures and not formulated in the earlier portion of the paper are the calculations of φ where:

$$\varphi = QH^{3/2} = (1 - e^{-GP})H^{3/2}/G \tag{12}$$

and:

$G = \alpha E'$, dimensionless materials parameter

α = pressure viscosity coefficient of lubricant

In the vector film shape code (figure 4) the expressions are written in vector form throughout and the IF Statement in figure 3 is completely eliminated. Vectorization requires the elimination of IF, CALL, and other interrupts within DO loops.

Results

Table I shows the scalar and vector computation times expressed in nanoseconds for varying vector length. A considerable decrease in computation time is seen to exist for the vector computation over that of the scalar computation. The last column of the table gives the ratio of scalar to vector computation times. This column indicates that as the vector length increases the scalar to vector time ratio increases significantly until the vector length becomes too large and an asymptotic condition is approached. The vector length used by Hamrock and Jacobson (1982) in evaluating elastohydrodynamic lubrication of rectangular contacts was 660. For this example, the table indicates that over a 50-to-1 speedup in the film shape calculations can be obtained on supercomputers that utilize vector computation. Figure 5 shows the results presented in Table I in graphical form. Here the leveling off of the scalar to vector computation times as the vector length becomes very large is quite apparent. In a typical elastohydrodynamic lubrication run the film shape code shown in figure 4 is used 2000 times.

Conclusions

A brief description of supercomputers' capabilities has been briefly presented. To demonstrate their usefulness in solving tribological problems, a simple kernel of the film shape calculations in an elastohydrodynamically lubricated rectangular contact was presented. Both scalar and vector versions of the film shape code were presented. The run times for the two types of code indicate that over a 50-to-1 speedup of scalar to vector computation time is obtained for vector lengths typically used in elastohydrodynamic lubrication analysis. The significance of these results should prove useful in considering additional aspects in elastohydrodynamic lubrication such as:

(1) Thermal effects
(2) Surface roughness effects
(3) Non-Newtonian effects of the fluid
(4) Side leakage effects

The use of supercomputers in this decade for the incorporation of these coupled effects should reduce run times considerably and thereby make such calculations more feasible.

References

1. Hamrock, B. J. and B. Jacobson (1982) Elastohydrodynamics Lubrication of Rectangular Contacts, submitted to be presented at the 1982 Joint Lubrication Conference and later published in JOLT of the American Society of Mechanical Engineers.

2. Levine, R. D., "Supercomputers," Sci. Am., 246, (1), 118-135 (1982).

3. Timoshenko, S. and Goodier, J. N., Theory of Elasticity, 2nd ed., McGraw Hill (1951).

TABLE I. - COMPARISON OF SCALAR AND VECTOR COMPUTER RUN TIMES
FOR DIFFERENT VECTOR LENGTHS

Vector length, NX	Scalar time, nsec	Vector time, nsec	$\frac{\text{Scalar time}}{\text{Vector time}}$
10	241	70	3.44
30	1 179	155	7.61
50	2 726	243	11.22
100	9 252	483	19.28
300	73 357	1 933	37.95
500	198 282	4 188	47.35
700	383 997	7 242	53.02
1000	776 588	13 363	58.11
3000	6 890 354	99 988	68.91
5000	19 084 530	266 602	71.58
7000	37 358 446	513 218	72.79
10000	76 171 860	1 033 169	73.73

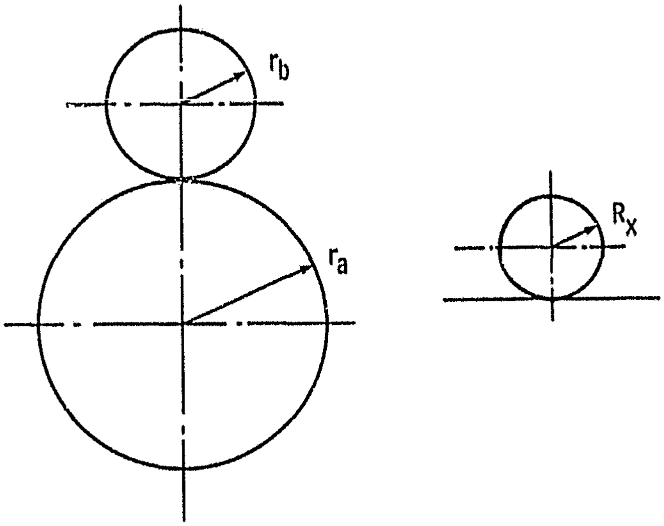

Figure 1. - Rollers and equivalent roller.

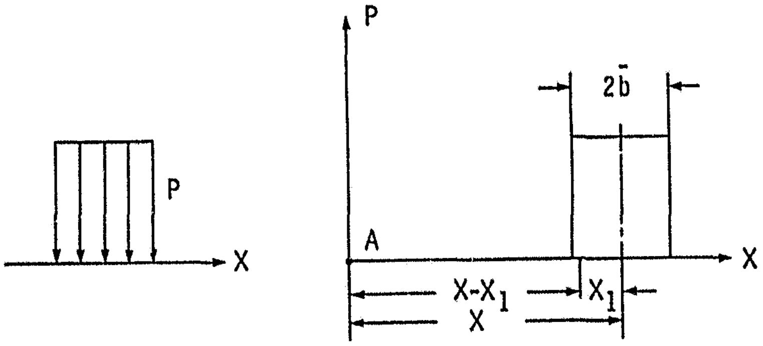

Figure 2. - Surface deformation of semi-infinite body subjected to uniform pressure over a rectangular area.

```
      DO 8 I = 1, NX
      SUM = .ODO
      DO 10 II = 1, NX
      M = IABS (I-II) + 1
   10 SUM = SUM + PR (II) *D(M)
      W(I) = 2.ODO*SUM/PI
      H(I) = HO1+RX*(S(I)+W(I))
      IF (H(I).GT.HMIN) GO TO 400
      HMIN = H(I)
      NSAVE = I
  400 CONTINUE
    8 PHI(I) = (1.ODO-DEXP(-G*PR(I)))*(H(I)**1.5ODO)/G
```

Figure 3. - Scalar film shape code.

```
C           GI = 1.0/G
C
            DO 500 I, NX
            DD(I) =D(NX+1-I)
        500 CONTINUE
C
            DO 510 I=1, NX
            DD(I+NX-1) = D(I)
            W(I) = PR(1)*DD(I+NX-1)
        510 CONTINUE
C
            DO 530 K = 2, NX
            DO 520 I = 1, NX
            W(I) = W(I)+PR(K)*DD(I+NX-K)
        520 CONTINUE
        530 CONTINUE
C
            DO 540 I = 1, NX
            W(I) = (2.0/PI)*W(I)
            H(I) = HO1+RX*(S(I)+W(I))
            PHI(I) = (1.0-EXP(-G*PR(I)))*(H(I)*SQRT(H(I)))*GI
        540 CONTINUE
C
            NSAVE = Q8SMINI(H(1;NX))+1
            HMIN = H(NSAVE)
C
```

Figure 4. - Vector film shape code.

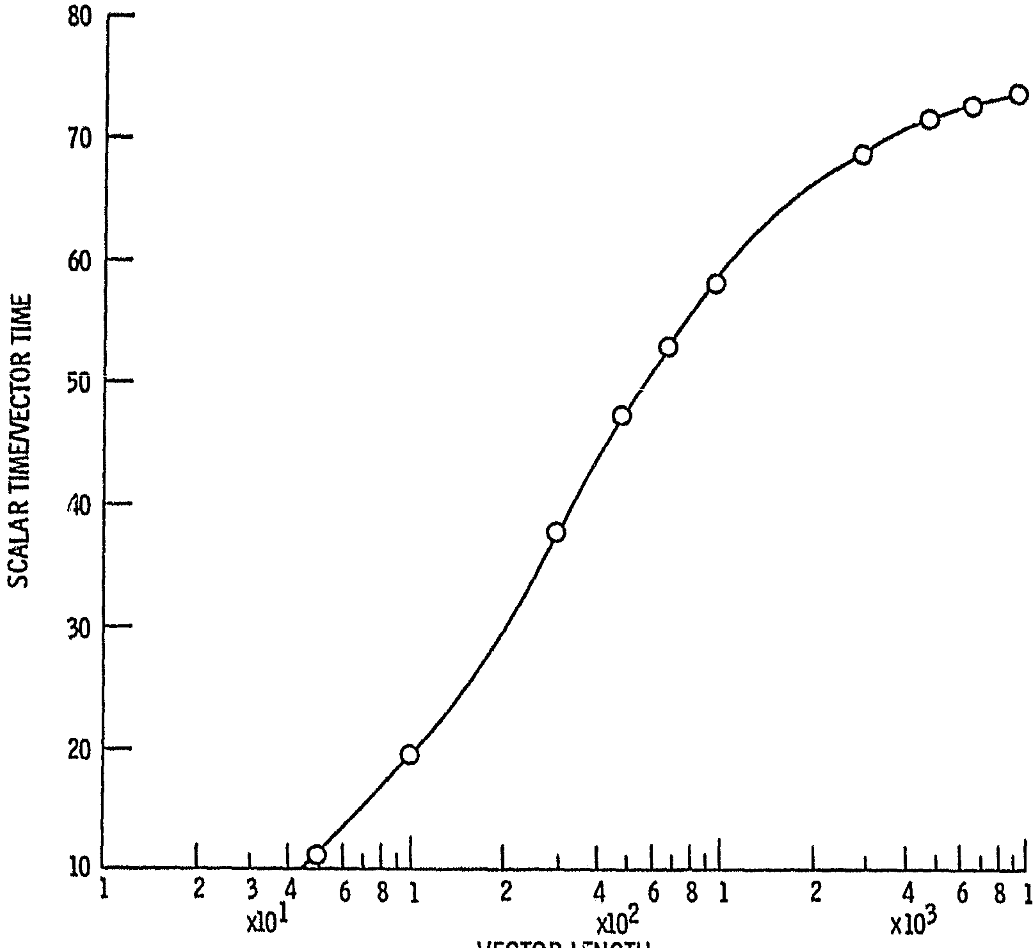

Figure 5. - Effect of scalar run time divided by vector run time on various vector lengths.