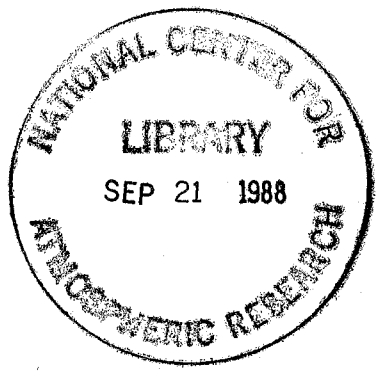
NCAR/TN-140+STR
NCAR TECHNICAL NOTE

June 1979

Mie Scattering Calculations: Advances in Technique and Fast, Vector-Speed Computer Codes

Warren J. Wiscombe



ATMOSPHERIC ANALYSIS AND PREDICTION DIVISION

NATIONAL CENTER FOR ATMOSPHERIC RESEARCH
BOULDER, COLORADO

ABSTRACT

Dave published the first widely-used Mie scattering codes in 1968. Even on the fastest computers, these codes and their descendants often took a great deal of computer time for problems of practical interest. In the intervening years, there have been a number of improvements in technique (some developed by this author and reported herein for the first time). At the same time, vector processing has increasingly become the wave of the future in scientific computing machines, and Mie computations can be effectively reorganized to take advantage of it.

The present document gathers these improvements in technique, together with the necessary reorganization to attain vector speed, to produce new Mie scattering codes suitable for modern computation.

Actually, two codes are presented. The first, MIEV0, attains as much vector speed as possible within the constraint of using the minimum possible memory. MIEV0 is suitable for almost any computer, whether or not it has vector capabilities, and is generally faster and better designed than existing Mie codes.

The second code, MIEV1, attains maximum vector speed, at the expense of using more memory than MIEV0. It is most suitable for vector-processing computers. MIEV1 is anywhere from 10% to 300% faster on the CRAY-1 than MIEV0.

Detailed timing results are presented for both codes. The codes are thoroughly tested and documented and are exceptionally reliable and well-conditioned. Mie results up to size parameters of 20,000 have been generated without difficulty.

PREFACE

My own investigations in radiative transfer in the atmosphere have always depended on Mie scattering calculations, and such calculations have invariably devoured the lion's share of my computing resources. Therefore, I continually sought ways to speed-up Mie scattering calculations (or circumvent them). No major breakthroughs along these lines appeared in the literature. In the meantime, I made several minor advances in the Mie algorithms, none of which I published.

Then came the new vector-processing computers (like NCAR's CRAY-1), with their promise of speed increases of tenfold or more if one "vectorized" one's calculations. With this incentive, I completely scrapped my old set of Mie codes and proceeded to write, from scratch, Mie codes that would work at vector speed, yet be compatible with scalar machines. They incorporate everything I have learned about Mie calculations, both by myself and from others, during the past seven years. Several colleagues who tested these codes found them significantly better than what they had been using before; in particular, they reported speed increases by factors of 6 to 100. Hopefully other investigators will find the routines as useful as they did.

Many thanks to Holly Howard for her excellent job in typing my rather unruly manuscript and to Dr. James Coakley for reviewing it.

Warren Wiscombe
June 1979

TABLE OF CONTENTS

1. INTRODUCTION

Calculations of light scattering from particles are needed in the widest variety of research endeavors, ranging from astronomy (interstellar dust scattering) to zoology (bacterial scattering). The prototypical such calculation assumes that the particle is a homogeneous sphere, and the incident light a monochromatic plane wave. Clebsch worked out the mathematical machinery for this problem in 1861; in 1890 Lorenz gave a full solution for transparent spheres; in 1908 Mie, and in 1909 Debye, published completely general solutions. These references, and many others up through 1945, may be found in the excellent historical survey by Logan (1965). By an odd twist of fate, Mie's name has come to be exclusively associated with the problem; we shall adhere to this convention, but with full awareness that a misnomer is involved. Excellent accounts of Mie theory may be found in the books by Shifrin (1951), Van de Hulst (1957), Kerker (1969) and Born and Wolf (1975).

While the vast majority of scattering particles are not spherical, so that the Mie solution does not strictly apply to them, both intuition and experimental evidence (e.g., Zerull, 1976) indicate that, with averaging over orientation and/or size, mildly nonspherical particles scatter very much like 'equivalent' spheres. This, of course, vastly enhances the utility of the Mie solution. Van de Hulst (1957) and Kerker (1969) indicate some of the wide-ranging applications which are then possible.

The computational history of Mie scattering is quite different and much more recent than the theoretical history; Van de Hulst's

and Kerker's books both contain accounts of it. Van de Hulst notes that, while there have been some ambitious tabulations of various Mie quantities, these are useful mostly for checking computer codes. The reason is simply the rapid oscillation of most Mie quantities; these oscillations would be impossible to resolve in a comprehensive table, and they make tabular interpolation risky. Furthermore, there are resonances (sharp spikes) within Mie quantities whose scale is much finer even than the oscillations (Rosasco and Bennett, 1978; Chýlek et al., 1978). Hence Mie scattering is a field which, unlike some, absolutely demands computer calculations. (Analytic approximations to Mie's solution are usually very limited in their range of application and, even when available, have an unfortunate tendency to be at least as complicated as the Mie solution itself.)

The number of Mie scattering calculations performed prior to the mid-1950s was expectably small. Things picked up rapidly thereafter, but until the late 1960s Mie calculations were done without much understanding of computational error, particularly ill-conditioning from subtraction of nearly equal numbers, and instability in recursion relations. Defining the size parameter

$$x = \frac{\text{circumference of sphere}}{\text{wavelength}}$$

we may easily find cases where a straightforward programming of Mie's solution would incur serious computational error for x as small as 20-30. Thus, while many of the pre-1970 Mie results are undoubtedly correct, it would be foolish to place blind faith in them.

Because the Mie solution is a series with approximately x terms, because early computers were slow, and because $x > 1$ was virtual terra incognita, early computer calculations were generally restricted to $1 < x < 100$. Irvine's 1965 paper is an excellent and frequently cited example from this period. But the explosive growth of computer technology soon permitted calculations for much larger size parameters; Dave's 1968b paper, which considers up to $x = 785$, is a good example.

A further development was Kattawar and Plass' (1967) paper pointing out the potential instability in the upward-recursive calculation of a Bessel function ratio in Mie theory (cf. Eq. 20). (Rayleigh actually discovered this around the turn of the century (Logan, 1965), but his observations had long been forgotten.) Their resolution of the difficulty, which has become standard, was to use downward recursion. Deirmendjian's (1969) book describes the same instability problem but offers no solution to it.

Dave's (1968a, 1969a) IBM reports were important landmarks. They set forth an algorithm and a brace of computer codes for performing Mie calculations. While the number of different Mie codes in use today is undoubtedly large, most of them can trace their lineage back to Dave's subroutines.

With powerful subroutines like Dave's in hand, the 1970s saw an explosive growth in applications. My own experience may be illustrative. In 1971 I began investigating solar and longwave radiation transfer in an atmosphere containing clouds. The presence of clouds required Mie scattering calculations for a size distribution of water droplets for 100 or more wavelengths from 0.3 μm to 500 μm. Consider a wavelength of 0.5 μm and a cloud with drop sizes ranging from 0.1 μm to 50 μm.

The Mie calculation for this case proceeds from x = 1 to x = 628, in steps $\Delta x = 0.1$ (Dave, 1969b); for each value of x there are roughly x terms to be summed, at each of say 100 angles, so the total number of terms to be summed is about 200 million. And each term by itself requires considerable computation to generate! Now imagine this calculation repeated for 100 wavelengths (it is less onerous for longer wavelengths) and it will become clear why 'staggering' is not an exaggerated description of the task. Yet, as the decade wore on, applications-oriented Mie calculations of this magnitude became increasingly commonplace.

In spite of this pressure of applications, the decade has so far brought forward no great changes in the algorithmic structure of Mie computations, save for Palmer's (1977) work suggesting that replacing series by continued fractions may be faster and more accurate. There have, however, been a few minor improvements. One is a new way to initialize the downward Bessel function ratio recursion (Lentz, 1976). Others are in the nature of more efficient ways to organize the computation, devised by the present author in response to the problem outlined in the previous paragraph and published here for the first time.

The present document has, as its overall purpose, the incorporation of these advances in technique, made in the decade since Dave (1968a, 1969a), into a new set of Mie scattering codes. The specific goals of this effort are laid out in the following section, and the analytical and computational details are worked out in the remainder of the document.

To the uninitiated reader, the codes may appear as if they could have been written straightaway, without all the apparent niggling over small points. But their simplicity is deceptive; it conceals a multitude of blind alleys, pitfalls, instabilities, inefficiencies and inelegancies to which the "straightaway" approach would fall victim. A variety of decisions has to be made, often requiring considerable background study and some sophistication in numerical analysis. Further advances will undoubtedly be forthcoming; but these algorithms already represent quite an advanced stage.

2. SPECIFIC GOALS OF ALGORITHMS

The specific goals that guided the formulation of the algorithms and codes herein were:

- maximum speed (including vector processing wherever possible)
- generality (furnishing all Mie quantities necessary for full polarization-dependent studies)
- reliability over broad ranges of refractive index and size parameter
- avoidance of all numerical instability
- portability (immediate executability on computers having single precision of at least 14 significant digits, like CDC and CRAY; executability with minor alterations on computers having significantly lower single precision)
- accuracy: at least 5-6 significant digits
- as simple and straightforward as possible within the constraints imposed by the previous goals.

Another goal, the use of minimum computer memory, proved incompatible in part with the main goal of maximum speed. This is because vector processing requires quantities, which could otherwise be scalars, to be stored in arrays. Therefore two separate codes were devised which have much in common but which either (a) have maximum speed, using as much memory as needed to achieve it (MIEV1); or (b) use minimum memory, attaining as much speed as possible within that constraint (MIEV0).

3. MIE SCATTERING FORMULAE--GENERAL CASE

The Mie scattering formulae are given in several books (Van de Hulst, 1957; Kerker, 1969; Deirmendjian, 1969) and by Dave (1968a, 1969a), although not always in the forms most suited to computation. We merely transcribe most of the relevant formulae below and give a reference; for those which are new, short derivations are provided. The five formulae (6-9) give the quantities actually calculated and returned by the codes MIEV0 and MIEV1.

The two important independent variables in Mie theory are the size parameter

$$x = \frac{2\pi r}{\lambda} \tag{1}$$

and

$$z = mx \tag{2}$$

where

r = radius of scattering sphere

λ = wavelength of incident plane wave

m = complex refractive index of sphere relative to surrounding medium

$$m = m_{re} - im_{im} \tag{3}$$

$$|m| > 1 \tag{4}$$

$$m_{im} > 0 \tag{5}$$

Restriction (4), which would, for example, exclude scattering from air bubbles in water, may not be necessary, but we have not tested the codes for such situations.

Extinction Efficiency Factor (Van de Hulst, Sec. 9.32)

$$Q_{ext} = \frac{2}{x^2} \sum_{n=1}^{N} (2n + 1) \text{ Re } (a_n + b_n) \tag{6}$$

Scattering Efficiency Factor (Van de Hulst, Sec. 9.32)

$$Q_{sca} = \frac{2}{x^2} \sum_{n=1}^{N} (2n + 1) \{|a_n|^2 + |b_n|^2\} \tag{7}$$

Asymmetry Factor (Kerker, Eq. 3.11.6)

$$g = \frac{4}{x^2 Q_{sca}} \sum_{n=1}^{N} \left\{ \frac{n(n + 2)}{n + 1} \text{ Re } (a_n a_{n+1}^* + b_n b_{n+1}^*) + \frac{2n + 1}{n(n + 1)} \text{ Re } (a_n b_n^*) \right\} \tag{8}$$

Scattering Amplitudes (Van de Hulst, Sec. 9.31)

$$S_1(\mu) = \sum_{n=1}^{N} \frac{2n + 1}{n(n + 1)} \{a_n \pi_n(\mu) + b_n \tau_n(\mu)\} \tag{9a}$$

$$S_2(\mu) = \sum_{n=1}^{N} \frac{2n + 1}{n(n + 1)} \{a_n \tau_n(\mu) + b_n \pi_n(\mu)\} \tag{9b}$$

where $\mu = \cos\theta$ and θ is the angle of scattering.

The most efficient way to calculate the scattering amplitudes is in a loop over n, for summing, within which is a loop over θ. (This will be explained when we discuss vectorization, Sec. 7.1.) Every extra operation in the θ-loop can add significantly to the computation time. Each term of the sums (9a) and (9b) requires two multiplies and one add, if the factor $(2n + 1)/n(n + 1)$ is incorporated into a_n and b_n outside the θ-loop. We discovered that we could

eliminate one of the multiplies by calculating, not S_1 and S_2 but rather,

$$S^+(\mu) = S_1 + S_2 = \sum_{n=1}^{N} \frac{2n+1}{n(n+1)} (a_n + b_n)\{\pi_n(\mu) + \tau_n(\mu)\} \tag{10a}$$

$$S^-(\mu) = S_1 - S_2 = \sum_{n=1}^{N} \frac{2n+1}{n(n+1)} (a_n - b_n)\{\pi_n(\mu) - \tau_n(\mu)\} \tag{10b}$$

The factors multiplying $(\pi_n \pm \tau_n)$ are formed outside the θ-loop. Then, when the n-loop is finished, S_1 and S_2 are easily recaptured:

$$S_1(\mu) = \frac{1}{2}[S^+(\mu) + S^-(\mu)] \tag{11a}$$

$$S_2(\mu) = \frac{1}{2}[S^+(\mu) - S^-(\mu)] \quad . \tag{11b}$$

If $\theta \in [0, \pi/2]$, then its supplement $\pi - \theta$ has cosine $-\mu$. The angular functions of $\pi - \theta$ are then simply related to the angular functions of θ (Dave, 1969a):

$$\pi_n(-\mu) = (-1)^{n+1} \tau_n(\mu)$$

$$\tau_n(-\mu) = (-1)^n \tau_n(\mu) \quad ,$$

which leads to

$$S^+(-\mu) = \sum_{n=1}^{N} (-1)^{n+1} \frac{2n+1}{n(n+1)} (a_n + b_n) \{\pi_n(\mu) - \tau_n(\mu)\} \tag{12a}$$

$$S^-(-\mu) = \sum_{n=1}^{N} (-1)^{n+1} \frac{2n+1}{n(n+1)} (a_n - b_n) \{\pi_n(\mu) + \tau_n(\mu)\} \tag{12b}$$

Thus it requires only two extra multiplies in the θ-loop to get $S^+(-\mu)$ and $S^-(-\mu)$, since the necessary quantities have already been formed in Eq. (10). For this reason, the codes are set up (as were Dave's, 1969a) to calculate the scattering amplitudes at an arbitrary set of

angles in $[0, \pi/2]$, but only at the supplements of those angles in $[\pi/2, \pi]$.

This is not as restrictive as it might seem. For example, in order to calculate the phase function moments, or other angular integrals over $[0, \pi]$, this is frequently the required structure of the angular quadrature point set. Also, when taking a dense mesh of angles near $\theta = 0$ to resolve the forward diffraction peak, one automatically obtains good resolution of the structure in the glory region. Last, but not least, taking an arbitrary set of angles in $[\pi/2, \pi]$ may significantly increase computation time, through the necessity of calculating $\pi_n(\mu)$ and $\tau_n(\mu)$ at those angles.

Note finally that, if one is only interested in the unpolarized approximation, as in many radiative transfer problems, the final step, Eq. (11), is unnecessary, for

$$|S_1|^2 + |S_2|^2 = |S^+|^2 + |S^-|^2 \quad .$$

However, the side-stepping of Eq. (11) is not explicitly provided for in the codes, since the savings in computer time would be relatively trivial.

Mie Coefficients (Dave, 1969a, Eqs. 2-3)

$$a_n = \frac{\left\{\frac{A_n(z)}{m} + \frac{n}{x}\right\}\psi_n(x) - \psi_{n-1}(x)}{\left\{\frac{A_n(z)}{m} + \frac{n}{x}\right\}\zeta_n(x) - \zeta_{n-1}(x)} \qquad (16a)$$

$$b_n = \frac{\left|mA_n(z) + \frac{n}{x}\right| \psi_n(x) - \psi_{n-1}(x)}{\left|mA_n(z) + \frac{n}{x}\right| \zeta_n(x) - \zeta_{n-1}(x)} \tag{16b}$$

Here we have used the Debye and Van de Hulst (1957, Sec. 9.22) notation for the Riccati-Bessel functions ψ_n and ζ_n, which are defined below along with the function A_n.

Ricatti-Bessel Functions (Abramowitz/Stegun, Sec. 10.3)

$$\psi_n(x) = xj_n(x)$$

$$\chi_n(x) = -xy_n(x)$$

$$\zeta_n(x) = \psi_n(x) + i\chi_n(x)$$

where j_n, y_n are the familiar spherical Bessel functions. The Riccati-Bessel functions satisfy the same recurrence relations as j_n, y_n (Abramowitz/Stegun, Eq. 10.1.19), namely

$$\zeta_{n+1}(x) = \frac{2n+1}{x} \zeta_n(x) - \zeta_{n-1}(x) \quad . \tag{17}$$

Furthermore, by a simple modification of the cross-product relation for spherical Bessel functions (Abramowitz/Stegun, Eq. 10.1.31), we find

$$\psi_{n+1}(x) = \frac{\psi_n(x)\, \chi_{n+1}(x) - 1}{\chi_n(x)} \quad . \tag{18}$$

The initial values are (Abramowitz/Stegun, Eq. 10.3.2-3)

$$\psi_o(x) = \sin x \tag{19a}$$

$$\chi_o(x) = \cos x \tag{19b}$$

$$\psi_1(x) = \frac{1}{x}\ \psi_o(x) - \chi_o(x) \qquad (19c)$$

$$\chi_1(x) = \frac{1}{x}\ \chi_o(x) + \psi_o(x) \quad . \qquad (19d)$$

Careful attention to the accuracy of the ψ_n, χ_n calculation is necessary because: first, the ψ_n up-recurrence is numerically unstable (Abramowitz/Stegun, Introduction .7); second, the convergence of the Mie series hinges on the rapid decay of ψ_n to zero when $n \gtrsim x$, with a concomitant blow-up of χ_n, which forces a_n and b_n (Eq. 16) to zero; third, phenomena like the glory, surface waves, and resonances depend on Mie series terms with $n \gtrsim x$, and hence are particularly sensitive to errors in ψ_n or χ_n.

Dave (1968a, 1969a) used upward recurrence (Eq. 17) for ζ_n, i.e., for both ψ_n and χ_n. He needed IBM double precision (equivalent to CDC or CRAY single precision) to minimize numerical instability, i.e., to minimize error growth relative to the magnitude of ψ_n as the recurrence proceeded. In our own early experiments on a UNIVAC 1108, we also found that up-recurrence on ψ_n deteriorated much too catastrophically for $n > x$ when single precision (8 digits) was used, and it was necessary to go over to UNIVAC double precision (18 digits) to achieve satisfactory accuracy.

We tested three possible schemes for computing ψ_n. (χ_n is always computed by upward recurrence, which is stable.) These were:

(a) upward recurrence (Eq. 17)

(b) cross product (Eq. 18)

(c) downward recurrence (Abramowitz/Stegun, Sec. 10.5) .

All computations were carried out in CDC single precision (14 digits). Scheme (c), which is stable, was used for determining errors in the other two schemes. It was initialized, in the manner of Miller, by

$$\psi_{N^*+1} = 0$$

$$\psi_{N^*} = 1$$

where N* exceeds the largest index N for which ψ_n is needed. We tried both N* = 1.9 N and N* = 1.5 N and obtained identical values of ψ_n (n = 1 to N) in either case, which satisfied us that the down-recurrence was an accurate and reliable benchmark.

Schemes (a) and (b) prove to be almost as accurate as down-recurrence for n < x (except as noted below). Only when n > x do they begin to deteriorate, and their errors grow monotonically as n continues to increase beyond x. Both schemes invariably have their largest error at n = N; these errors are shown in Table 1 for a wide range of x's. Up-recurrence is always more accurate than the cross-product relation; typically, its error is two to four times smaller. This situation holds for n < N as well.

Furthermore, up-recurrence has an <u>acceptable</u> error, as Table 1 witnesses (remember Table 1 shows <u>worst</u> errors; errors are much smaller for n < N). And since down-recurrence requires considerably more computation, more storage, and is also subject to overflow failure (because ψ_n <u>grows</u> when recursed downward), up-recurrence on ψ_n was the obvious choice for our algorithms. Thus, in the end, we made the same choice as Dave, but our confidence in that choice was considerably raised.

Table 1. Percent error in $\psi_N(x)$, as computed by two different methods, where N is the largest index required in the Mie series. Exact values of ψ_N were from downward recurrence. All computations done in CRAY-1 single precision.

x	N	% error in ψ_N from up-recurrence	% error in ψ_N from cross-product
1	5	1.7×10^{-5}	2.5×10^{-5}
5	12	8.2×10^{-5}	17×10^{-5}
10	19	2.0×10^{-4}	-13×10^{-4}
20	31	2.7×10^{-4}	5.6×10^{-4}
40	54	7.2×10^{-4}	15×10^{-4}
80	98	1.9×10^{-3}	6.0×10^{-3}
200	224	1.4×10^{-3}	4.5×10^{-3}
400	430	2.4×10^{-3}	5.5×10^{-3}
600	634	5.6×10^{-4}	266×10^{-4}
800	837	2.6×10^{-3}	3.5×10^{-3}
1,000	1,040	3.6×10^{-3}	10×10^{-3}
2,000	2,050	5.5×10^{-3}	15×10^{-3}
4,000	4,063	0.010	0.047
8,000	8,080	0.026	0.090
12,000	12,091	0.060	0.095
16,000	16,100	0.041	-0.17

There are also occasional values of $n < x$ at which the accuracy of both schemes (a) and (b) is reduced. These are invariably where ψ_n falls several orders of magnitude below its neighboring values ψ_{n-1} and ψ_{n+1}. In all such cases, up-recurrence is still preferable to the cross-product relation and still has acceptable error.

Since the zero-subscript feature is not widely available in FORTRAN, the quantities actually calculated in the codes are $\hat{\psi}_n = \psi_{n-1}$ and $\hat{\zeta}_n = \zeta_{n-1}$ (χ_n is never used explicitly).

A_n (Logarithmic Derivative of ψ_n)

$$A_n(z) \equiv \frac{\psi_n'(z)}{\psi_n(z)} \quad .$$

This can be calculated by up-recurrence (Dave, 1969a)

$$A_n(z) = -\frac{n}{z} + \frac{1}{\frac{n}{z} - A_{n-1}(z)} \qquad (n = 2,\ldots,N) \tag{20}$$

with initial value

$$A_1(z) = -\frac{1}{z} + \frac{\tan z}{\frac{1}{z}\tan z - 1} \quad . \tag{21}$$

Dave worried about two problem cases. First, $z = k\pi$ for any integer $k \neq 0$ leads to $A_o = \cot z = \infty$; but we never actually need A_o, and A_1, written as in Eq. (21), is perfectly well-conditioned at $z = k\pi$. Second, if tan z is written simplistically as sin z/cos z, exp(Im(z)) will occur in both numerator and denominator, causing overflow if Im(z) is large; but we circumvent this problem by writing tan z as

$$\tan(x + iy) = \frac{2e^{2y}\sin 2x + i(e^{4y} - 1)}{2e^{2y}\cos 2x + 1 + e^{4y}}$$

which is calculated by the function CTAN listed in Appendix I. There remains the problem that tan z blows up when z is an odd multiple of $\pi/2$, but because of round-off in computing the denominator this never actually occurs.

Kattawar and Plass (1967) pointed out the instability in up-recurrence of A_n when Im(z) is appreciable. They suggested down-recurrence

$$A_{n-1}(z) = \frac{n}{z} - \frac{1}{\frac{n}{z} + A_n(z)} \quad (n = N', \ldots, N, \ldots, 2) \tag{22}$$

as an alternative, and showed that it was stable. However, neither they nor subsequent investigators offered a clear-cut criterion as to when down-recurrence should be used. We have developed such a criterion, which is described in Sec. 5.

Kattawar and Plass, Dave (1969a), and many subsequent investigators initialized the down-recurrence by

$$A_{N'}(z) = 0$$

where, in principle, one must have $N' \gg |z|$. Dave found that

$$N' = 1.1\,|z| + 1$$

was sufficient. Of course, this causes more values of A_n to be calculated than are actually needed in the Mie series, since $N' > N$ and possibly $N' \gg N$.

We found it preferable to initialize the down-recurrence by calculating $A_N(z)$ correctly to 5 or 6 significant digits, using the newly-developed method of Lentz (1976) discussed below. Besides being more aesthetic, this results in less computation and greater

code reliability, especially in difficult cases (e.g., $|m| \gg 1$ or $|m - 1| \ll 1$).

We note in passing that our early experiments on a UNIVAC 1108 showed that neither up- nor down-recurrence of A_n was sufficiently accurate in its 8-digit single precision. The down-recurrence sometimes deteriorated badly as $n \to 1$. UNIVAC double precision of 18 digits had to be employed.

A_N from Lentz Method

The standard continued fraction representation of A_N, which follows directly from the down-recurrence (22), is

$$A_N(z) = \lim_{i \to \infty} [a_1, a_2, \ldots, a_i] \tag{23}$$

where

$$[a_1, a_2, \ldots, a_i] = a_1 + \frac{1}{a_2 +} \cdots + \frac{1}{a_i} \tag{24}$$

$$a_1 = (N + 1)/z \tag{25a}$$

$$a_k = (-1)^{k+1} (2N + 2k - 1)/z \quad (k = 2,3,\ldots) \ . \tag{25b}$$

But this is really no different from Dave's procedure; setting $A_{N'} = 0$ amounts to setting $A_N = [a_1, a_2, \ldots, a_{N'-N}]$. Thus many terms are frequently required for convergence of the continued fraction (24), and recursive computation is impossible because $[a_1, \ldots, a_{i+1}]$ is not simply related to a few prior values of $[a_1, \ldots, a_i]$ but must be computed completely anew.

Lentz has discovered a product representation of Eq. (24) which can be computed recursively, and which requires, in just those cases

where down-recurrence of A_n is necessary, far fewer steps than Dave's method:

$$[a_1,\ldots,a_i] = \prod_{k=1}^{i} T_k \tag{26}$$

where

$$T_k = \begin{cases} a_1 & k = 1 \\ N_k/D_k & k > 1 \end{cases} \tag{27}$$

$$N_k = [a_k,\ldots,a_1] \tag{28a}$$

$$D_k = [a_k,\ldots,a_2] \tag{28b}$$

and where, by definition,

$$[a_j, a_j] \equiv a_j \ .$$

Unlike Eq. (24), Eq. (26) is well-suited to recursive computation in that

$$[a_1,\ldots,a_i] = [a_1,\ldots,a_{i-1}]\, T_i \tag{29}$$

and both the numerator and denominator of T_i follow immediately from the numerator and denominator, respectively, of T_{i-1}:

$$N_i = a_i + \frac{1}{N_{i-1}} \tag{30a}$$

$$D_i = a_i + \frac{1}{D_{i-1}} \ . \tag{30b}$$

We deem the iteration to have converged when $T_i = 1$ to a certain degree of accuracy, i.e., when

$$|\mathrm{Re}(T_i) - 1| < \varepsilon_2 \quad \underline{\text{and}} \quad |I_m(T_i)| < \varepsilon_2 \tag{31}$$

where we generally take $\varepsilon_2 = 10^{-8}$ and, as a result, obtain at least 5 to 6 significant figures in A_N.

It may occur that

$$\left|\frac{N_i}{a_i}\right| < \varepsilon_1 \quad \underline{\text{or}} \quad \left|\frac{D_i}{a_i}\right| < \varepsilon_1 \tag{32}$$

where $\varepsilon_1 << 1$ (we generally take $\varepsilon_1 = 10^{-2}$); this means that Eq. (30) has resulted in loss of significant digits in N_i and/or D_i, which would be magnified in N_{i+1} and/or D_{i+1}. In this case, we can sidestep the ill-conditioning by skipping the convergence test (31) and striding two iterations instead of one; i.e., we go immediately to

$$[a_1,\ldots,a_{i+1}] = [a_1,\ldots,a_{i-1}]\ T_i\ T_{i+1} \tag{33}$$

where

$$T_i\ T_{i+1} = \frac{\xi_1}{\xi_2} \tag{34a}$$

$$\xi_1 \equiv 1 + a_{i+1}\ N_i \tag{34b}$$

$$\xi_2 \equiv 1 + a_{i+1}\ D_i \ . \tag{34c}$$

Unlike the ratios T_i or T_{i+1} individually, their product (34a) is well-conditioned. In order to re-start the iteration (30) without requiring the ill-conditioned results N_{i+1} or D_{i+1} we can use Eq. (30) twice to obtain

$$N_{i+2} = a_{i+2} + \frac{1}{a_{i+1} + \frac{1}{N_i}} = a_{i+2} + \frac{N_i}{\xi_1} \tag{35a}$$

$$D_{i+2} = a_{i+2} + \frac{D_i}{\xi_2} \ . \tag{35b}$$

In Table 2 is shown, for selected values of m and x, the number of iterations of Lentz's method necessary to achieve convergence, i.e., the value of i such that condition (31) is satisfied for either $\varepsilon_2 = 10^{-6}$ or $\varepsilon_2 = 10^{-8}$. Note that, for $x \geq 100$ and Im(m) small enough, the number of iterations is roughly $|m - 1|x$. This is about the same number of iterations required by Dave's method $[(1.1|m|-1)x]$. But these cases of small Im(m) can be handled by up-recurrence (see Sec. 5). And for larger values of Im(m), Lentz's method requires far fewer iterations than Dave's.

Angular Functions

$$\pi_n(\mu) \equiv P_n'(\mu)$$
$$\tau_n(\mu) \equiv \mu\pi_n(\mu) - (1 - \mu^2)\,\pi_n'(\mu) \qquad (36)$$

(Dave, 1969a) where P_n is a Legendre polynomial. These functions can most simply be calculated by upward recurrence, which is numerically stable (Abramowitz/Stegun, Introduction.7). Since these recurrences are buried in the θ-loop where $S^{\pm}$ (Eq. 10) are calculated, formulating them more efficiently can lead to a substantial saving of computer time. Dave (1969a) gives particular forms for these recurrences; but we have discovered better ones which we derive below.

The usual 3-term recurrence relation for Legendre polynomials is

$$(n + 1)\,P_{n+1}(\mu) = (2n + 1)\,\mu P_n(\mu) - nP_{n-1}(\mu) \quad .$$

Differentiating this, and using the relation

$$\pi_{n+1}(\mu) - \pi_{n-1}(\mu) = (2n+1)\,P_n(\mu)$$

Table 2. Number of Lentz method iterations (Eq. 29) necessary to converge to $A_N(mx)$ for a range of size parameters x and refractive indices m. First figure refers to $\varepsilon_2 = 10^{-6}$, second (in parentheses) to $\varepsilon_2 = 10^{-8}$.

x	N	m	number of Lentz method iterations to get $A_N(mx)$
1	5	$1.05 - \{10^{-6} - 1\}i$	4 (5)
		$1.95 - \{10^{-6} - 1\}i$	5 (6)
10	19	$1.05 - \{10^{-6} - 1\}i$	7-8 (8-10)
		$1.95 - \{10^{-6} - 1\}i$	11-12 (14)
100	119	$1.05 - \{10^{-6} - 1\}i$	13 (16)
		$1.50 - \{10^{-6} - 10^{-2}\}i$	51-52 (55-57)
		$1.50 - 0.1\,i$	40 (46)
		$1.50 - i$	18 (22)
		$1.95 - \{10^{-6} - 10^{-2}\}i$	97-99 (102-103)
		$1.95 - 0.1\,i$	79 (87)
		$1.95 - i$	25 (31)
1,000	1,040	$1.05 - \{10^{-6} - 10^{-3}\}i$	59-50 (58)
		$1.05 - 10^{-2}i$	44 (53)
		$1.05 - 0.1\,i$	25 (32)
		$1.05 - i$	15 (20)
		$1.50 - \{10^{-6} - 10^{-3}\}i$	500-502 (509-512)
		$1.50 - 10^{-2}\,i$	459 (479)
		$1.50 - 0.1\,i$	104 (135)
		$1.50 - i$	22 (28)
		$1.95 - \{10^{-6} - 10^{-3}\}i$	953-956 (964-967)
		$1.95 - 10^{-2}\,i$	880 (919)
		$1.95 - 0.1\,i$	203 (254)
		$1.95 - i$	33 (42)
10,000	10,086	$1.05 - \{10^{-6} - 10^{-4}\}i$	492-495 (510-513)
		$1.05 - 10^{-2}\,i$	183 (232)
		$1.05 - 0.1\,i$	31 (40)
		$1.05 - i$	16 (20)
		$1.50 - \{10^{-6} - 10^{-4}\}\,i$	4496-5001 (5017-5021)
		$1.50 - 10^{-2}\,i$	1092 (1384)
		$1.50 - 0.1\,i$	124 (161)
		$1.50 - i$	23 (30)
		$1.95 - \{10^{-6} - 10^{-4}\}i$	9501-9510 (9525-9533)
		$1.95 - 10^{-2}\,i$	2055 (2599)
		$1.95 - 0.1\,i$	236 (309)
		$1.95 - i$	34 (44)

from Whittaker and Watson (1965, chapter 15) leads to

$$n\pi_{n+1}(\mu) = (2n + 1)\,\mu\pi_n(\mu) - (n + 1)\,\pi_{n-1}(\mu)$$

which for our own purposes we write in the strung-out form

$$s \equiv \mu\pi_n(\mu) \tag{37a}$$

$$t \equiv s - \pi_{n-1}(\mu) \tag{37b}$$

$$\pi_{n+1}(\mu) = s + \left(\frac{n + 1}{n}\right) t \quad . \tag{37c}$$

Written thus, only two multiplies and two adds are required, since the purely numerical factor (n+1/n) may be precalculated.

For τ_n, differentiate the relation

$$(1 - \mu^2)\,\pi_n(\mu) = n[P_{n-1}(\mu) - \mu P_n(\mu)]$$

and use the relation

$$nP_n(\mu) = \mu\pi_n(\mu) - \pi_{n-1}(\mu)$$

(both from Whittaker/Watson, chapter 15) to obtain

$$(1 - \mu^2)\,\pi_n'(\mu) = 2\mu\pi_n(\mu) - (n + 1)\,[\mu\pi_n(\mu) - \pi_{n-1}(\mu)] \quad .$$

Putting this into the definition (36) of τ_n leads to

$$\tau_n(\mu) = (n + 1)\,t - s \tag{38a}$$

$$= nt - \pi_{n-1}(\mu) \tag{38b}$$

where t and s were defined in Eq. (37). This requires only one multiply and one add, for a total of three multiplies and three adds to calculate π_{n+1} and τ_n. This compares with six multiplies and four adds in Dave's (1969a) recurrences.

The recurrences are initialized by

$$\pi_0 = 0 \tag{39a}$$

$$\pi_1 = 1 \ . \tag{39b}$$

Note that we actually require, not π_n and τ_n separately, but their sum and difference, to obtain $S_1^{\pm}$ (Eq. 10). Using the definition (36) of τ_n,

$$\boxed{\pi_n \pm \tau_n = (1 \pm \mu)\,[(\mu \mp 1)\,\pi_n]' \ .}$$

Yet, in spite of this simple form, we were unable to come up with a recurrence directly for $\pi_n \pm \tau_n$, which avoided calculating π_n and τ_n individually, and was at the same time more efficient than Eqs. (37-38). This must remain a challenge for future investigators.

By using a fixed set of angles μ_m, one could precalculate the angular functions once for all, as matrices $A_{nm} \equiv \pi_n(\mu_m)$ and $B_{nm} \equiv \tau_n(\mu_m)$. This might offer advantages if one were to restrict oneself to Mie series no longer than $\sim$ 1000 terms and to no more than 50-100 angles. But outside these limits the computer storage requirements would be prohibitive; and even with only 100 terms and 50 angles, A_{nm} and B_{nm} would take up a total of 10,000 words of storage, which might well strain a smaller computer. Furthermore, a fixed set of angles would be too inflexible for many applications. Thus, in the interests of a general purpose code, we have rejected this approach.

4. MIE SCATTERING FORMULAE—SMALL PARTICLE LIMIT

It is necessary to compute the small-particle (Rayleigh) case, $x \to 0$, separately, not only because $1/x$ occurs in several places in the Mie formulae, but because:

(a) the calculation of A_n is numerically ill-conditioned as $x \to 0$; e.g.,

$$A_1(z) = -\frac{1}{z} + \frac{1}{\frac{1}{z} - \frac{1}{\tan z}} = \frac{2}{z} + O(z)$$

and the subtraction of $1/\tan z$ from $1/z$ causes significant digits to be lost; similarly,

$$A_2(z) = -\frac{2}{z} + \frac{1}{\frac{2}{z} - A_1(z)}$$

and the subtraction of $A_1(z)$ from $2/z$ further compounds the ill-conditioning.

(b) The subtraction in the upward recurrence for ψ_n loses significant digits, and the ill-conditioning rapidly compounds; e.g.,

$$\psi_1(x) = \frac{1}{x}\sin x - \cos x = \left(1 - \frac{x^2}{6} + \ldots\right) - \left(1 + \frac{x^2}{2} + \ldots\right)$$
$$= \frac{x^2}{3} + \ldots$$

$$\psi_2(x) = \frac{3}{x}\psi_1(x) - \psi_o(x) = \frac{3}{x}\left(\frac{x^2}{3} + \ldots\right) - \sin x$$
$$= (x + \ldots) - (x + \ldots) \quad .$$

(c) The subtraction in the numerator of b_n becomes ill-conditioned.

The $x \to 0$ formulae are also computationally faster, which would make them desirable even were they unnecessary. We should bear in mind Van de Hulst's (1957, Sec. 10.3) warning, however:

"Aside from their simplicity, they (the small-x formulae) have little advantage. They describe the very first deviations from Rayleigh scattering, but further deviations appear very soon after the first have become prominent, so that the full Mie formulae have to be used."

In view of this warning, we set ourselves the modest goal of applying the $x \to 0$ formulae only over a range of x sufficient to avoid serious ill-conditioning and to give six significant digits in all Mie quantities. The $x \to 0$ formulae which we found most suitable, and the range over which they are to be applied, are given below.

The obvious approach is to expand the Mie coefficients (Eq. 16) in powers of the size parameter x. It is necessary to expand the dominant coefficient a_1 at least out to $O(x^6)$, or else the extinction efficiency (Eq. 6) for $Im(m) \ll 1$ will be vanishingly small; for the first nonnegligible contribution to Q_{ext} is $O(x^6)$ in such cases. But by carrying a_1 out to $O(x^6)$, one must for consistency keep a_2 and b_1 also, for they are both $O(x^5)$. The remaining values of a_n and b_n are $O(x^7)$ or higher.†

Some authors insert the expansions for a_1, a_2 and b_1 into Eqs. (6-9) and continue expanding and truncating those series. This is useful for

---

†The asymptotic forms as $x \to 0$ are:

$$a_n \sim i \frac{n+1}{(2n-1)!!(2n+1)!!} \frac{m^2-1}{nm^2+n+1} x^{2n+1}$$

$$b_n \sim i \frac{m^2-1}{(2n+3)[(2n+1)!!]^2} x^{2n+3} .$$

seeing the leading terms, but every additional step of expansion/truncation must cause additional error, so we have chosen instead to use Eqs. (6-9) as is, namely:

$$Q_{ext} = 6x \{Re(\hat{a}_1 + \hat{b}_1) + \frac{5}{3} Re(\hat{a}_2)\} \tag{40a}$$

$$Q_{sca} = 6x^4 T \tag{40b}$$

$$g = Re[\hat{a}_1(\hat{a}_2 + \hat{b}_1)^*]/T \tag{40c}$$

$$S_1(\mu) = \frac{3}{2} x^3 [\hat{a}_1 + (\hat{b}_1 + \frac{5}{3} \hat{a}_2)\mu] \tag{40d}$$

$$S_2(\mu) = \frac{3}{2} x^3 [\hat{b}_1 + \hat{a}_1\mu + \frac{5}{3} \hat{a}_2(2\mu^2 - 1)] \tag{40e}$$

where

$$T \equiv |\hat{a}_1|^2 + |\hat{b}_1|^2 + \frac{5}{3} |\hat{a}_2|^2 \tag{40f}$$

and where a factor of x^3 has been taken out of each coefficient,

$$\hat{a}_1 \equiv \frac{a_1}{x^3} , \quad \text{etc.}$$

in order to avoid $\frac{0}{0}$ singularities as $x \to 0$.

An additional advantage of Eqs. (40) is that, unlike many of the expansions one finds in the literature, they are internally consistent, being simply special cases of the Mie formulae with $a_3 = a_4 = \ldots = b_2 = b_3 = \ldots = 0$. For example, Eqs. (40a-c) can be derived directly from Eqs. (40d,e).

Comparisons were made between the exact Mie results and Eqs. (40), using two different approximations for $\hat{a}_1$, $\hat{a}_2$ and $\hat{b}_1$, for a wide range of values of x and m. First, we tried the 3-term expansion of a_1 and the 1-term expansions of a_2 and b_1, as given by Van de Hulst (1957, Sec. 10.3), to wit,

$$\hat{a}_1 = i\frac{2}{3}\frac{m^2-1}{m^2+2}\left[1+\frac{3}{5}\frac{m^2-2}{m^2+2}x^2 - i\frac{2}{3}\frac{m^2-1}{m^2+2}x^3 + O(x^4)\right] \tag{41a}$$

$$\hat{b}_1 = i\frac{m^2-1}{45}x^2 + O(x^4) \tag{41b}$$

$$\hat{a}_2 = \frac{i}{15}\frac{m^2-1}{2m^2+3}x^2 + O(x^4) \quad . \tag{41c}$$

(Errors may still be found in such results; e.g., Kerker (1969, Sec. 3.9) has the wrong sign for b_1.) These were found to give our desired 6-digit accuracy for all Mie quantities, provided Im(m) was large enough. But when Im(m) << 1, the accuracy of Q_{ext}, $Re(S_1)$ and $Re(S_2)$ deteriorated to as little as 3 digits even for very small values of x. Examples of such errors are shown, for Q_{ext}, in Table 3. The remaining quantities, like Q_{sca} or $Im(S_1)$, retained 6-digit accuracy out to $x \sim 0.08$.

We diagnosed the problem to be that Q_{ext}, $Re(S_1)$ and $Re(S_2)$ all depend only on the real parts of $\hat{a}_1$, $\hat{a}_2$ and $\hat{b}_1$. For Eqs. (41), with Im(m) << 1, only the smallest-order term in $\hat{a}_1$ contributes to $Re(\hat{a}_1)$, and both $Re(\hat{a}_2)$ and $Re(\hat{b}_1)$ are nearly zero. Clearly a correction term would be desirable in $Re(\hat{a}_1)$, and presumably in $\hat{a}_2$ and $\hat{b}_1$ as well. Such a correction term can only be obtained by continuing the expansion of $\hat{a}_1$ out to $O(x^5)$. We also realized that it would be preferable to leave $\hat{a}_1$ in quotient form, that is, to just expand its numerator and denominator in Eq. (16). The further expansion of the quotient as a polynomial is just an extra and unnecessary step of approximation. Hence we used the following formulation:

Table 3. Values of Q_{ext} in the small-particle limit, as given by Mie theory and by Eq. (40) with two approximations for a_1, a_2 and b_1. The digits in error are underscored for each approximation.

x	m	Eqs. (40,41)	Eqs. (40,42)	Exact
0.02	$1.5 - 10^{-6}i$	7.67794×10^{-8}	7.67805×10^{-8}	7.67805×10^{-8}
	$1.95 - 10^{-6}i$	1.27341×10^{-7}	1.27355×10^{-7}	1.27355×10^{-7}
	$1.95 - 10^{-5}i$	3.77644×10^{-7}	3.77659×10^{-7}	3.77659×10^{-7}
0.04	$1.05 - 10^{-6}i$	1.12183×10^{-7}	1.12179×10^{-7}	1.12179×10^{-7}
	$1.5 - 10^{-6}i$	6.70337×10^{-7}	6.70403×10^{-7}	6.70403×10^{-7}
	$1.5 - 10^{-4}i$	8.57001×10^{-6}	8.57007×10^{-6}	8.57008×10^{-6}
	$1.95 - 10^{-4}i$	7.16164×10^{-6}	7.16259×10^{-6}	7.16259×10^{-6}
0.08	$1.05 - 10^{-6}i$	3.28743×10^{-7}	3.28478×10^{-7}	3.28478×10^{-7}
	$1.5 - 10^{-6}i$	9.60869×10^{-6}	9.61291×10^{-6}	9.61292×10^{-6}
	$1.5 - 10^{-4}i$	2.54505×10^{-5}	2.54547×10^{-5}	2.54547×10^{-5}
	$1.95 - 10^{-4}i$	3.66727×10^{-5}	3.67335×10^{-5}	3.67336×10^{-5}
0.20	$1.05 - 0.01i$	5.25292×10^{-3}	5.25256×10^{-3}	5.25263×10^{-3}
	$1.05 - i$	5.79042×10^{-1}	5.78532×10^{-1}	5.78539×10^{-1}
	$1.95 - 0.01i$	3.88981×10^{-3}	3.90526×10^{-3}	3.90548×10^{-3}
	$1.95 - i$	2.58547×10^{-1}	2.58626×10^{-1}	2.58637×10^{-1}

$$\hat{a}_1 = 2i \frac{m^2 - 1}{3} \frac{N_1}{D_1} \tag{42a}$$

$$N_1 \equiv 1 - \frac{1}{10} x^2 + \frac{4m^2 + 5}{1400} x^4 + O(x^6)$$

$$D_1 \equiv m^2 + 2 + (1 - \frac{7m^2}{10}) x^2 - \frac{8m^4 - 385m^2 + 350}{1400} x^4 + 2i \frac{m^2 - 1}{3} x^3 (1 - \frac{1}{10} x^2) + O(x^6)$$

$$\hat{b}_1 = i \frac{m^2 - 1}{45} x^2 \frac{1 + \frac{2m^2 - 5}{70} x^2 + O(x^4)}{1 - \frac{2m^2 - 5}{30} x^2 + O(x^4)} \tag{42b}$$

$$\hat{a}_2 = i \frac{m^2 - 1}{15} x^2 \frac{1 - \frac{1}{14} x^2 + O(x^4)}{2m^2 + 3 - \frac{2m^2 - 7}{14} x^2 + O(x^4)} . \tag{42c}$$

To be consistent, and because the amount of extra computation is trivial, we have expanded $\hat{b}_1$ and $\hat{a}_2$ to the same order as $\hat{a}_1$, even though neither one acquires thereby a significant real part when $\mathrm{Im}(m) \ll 1$, and therefore neither one significantly improves the approximation to Q_{ext} in such cases. But we have ignored $\hat{b}_2$ and $\hat{a}_3$, the leading terms of whose expansions contribute at the $O(x^4)$ level; these leading terms are purely imaginary when $\mathrm{Im}(m) = 0$, and therefore do not improve the approximation to Q_{ext}. Carrying $\hat{b}_2$ and $\hat{a}_3$ only extends the range of the approximation slightly (cf. the Van de Hulst quote earlier in this section) and simply would not be worthwhile.

Even though the neglect of $\hat{b}_2$ and $\hat{a}_3$ is a technical inconsistency, the practical effect is nil. As Table 3 indicates, Eqs. (42) lead to a dramatic improvement in accuracy over Eqs. (41) in both problem ($\mathrm{Im}(m) \ll 1$) and normal cases. The more accurate formulation retains

4-5 significant digits at $x = 0.2$, and 2-3 significant digits even at $x = 0.5$. In order to retain 6 significant digits, we selected the rather conservative criterion that the small-particle formulae (Eqs. 40 and 42) are to be used whenever

$$|m| \; x \leq 0.1 \tag{43}$$

where the factor $|m|$ was included based on the well-known fact (cf. Kerker et al., 1978), which is also apparent in Eqs. (42), that the approximation breaks down as $|m|$ increases.

5. A_n UP-RECURRENCE CRITERION

Kattawar and Plass (1967), Dave (1969a) and subsequent investigators were aware that up-recurrence for A_n may fail, but were unclear as to exactly <u>when</u> this would occur, except that Im(m) had to be significant and $x >> 1$ (the latter condition ensures sufficiently many iterations for the instability to develop).

To determine the precise regions of size parameter x and refractive index m wherein up-recurrence is satisfactory, an extensive comparison was made between exact Mie results and results obtained using up-recurrence for A_n. Exact Mie results were generated using down-recurrence for A_n, which is always stable, and a stringent convergence criterion of $\varepsilon_2 = 10^{-10}$ for the Lentz method calculation of A_N. Up-recurrence was regarded as failing whenever its concomitant values of Q_{ext}, Q_{sca} or g (Eqs. 6-8) had relative errors exceeding 10^{-6}, <u>or</u> its scattered intensity or degree of polarization

$$i_1 + i_2 = |S_1|^2 + |S_2|^2 \tag{44}$$

$$d_{pol} = \frac{|S_2|^2 - |S_1|^2}{|S_2|^2 + |S_1|^2} \tag{45}$$

had relative errors exceeding 10^{-5} at any of 61 Lobatto quadrature angles. (These angles are such that they cluster in the forward peak ($\theta \gtrsim 0°$) and glory ($\theta \lesssim 180°$) regions and are relatively more sparse around $\theta = 90°$.)

In practice, it was always these angular functions which first heralded failure; the relative errors in Q_{ext}, Q_{sca} and g were always between 10^{-12} and 10^{-8} when some angular function error reached 10^{-5}.

This is because there is no cancellation in the Q_{ext}, Q_{sca}, g series, whereas at some angles the $n < x$ terms in the S_1, S_2 series almost cancel, leaving only the $n > x$ terms, which are most affected by A_n up-recurrence instability.

The calculation was structured in the form of an upward search on m_{im}, for fixed m_{re} and x, to determine the smallest value, m_{im}^{crit}, such that up-recurrence on A_n failed. (Preliminary study showed that, when failure occurs for $m_{im} = m_{im}^{crit}$, it continues to occur for all $m_{im} > m_{im}^{crit}$.) The search for m_{im}^{crit} was successively refined to pinpoint it to 3 significant digits. Size parameter was varied in the range $1 \leq x \leq 10{,}000$ and m_{re} was varied from 1.05 to 2.50 in steps of 0.05.

It very quickly became evident that, for fixed m_{re}, as x increased the product xm_{im}^{crit} rapidly approached, from above, an asymptote. Thus there was a function $f(m_{re})$ such that

$$xm_{im}^{crit} \geq \min_x(xm_{im}^{crit}) \equiv f(m_{re}) \tag{46}$$

and such that the inequality was, in fact, roughly an equality over almost the entire range of x. The data for $\min_x(xm_{im}^{crit})$ as a function of m_{re} are given in Table 4 and plotted in Fig. 1.

It is plain from Fig. 1 that $f(m_{re})$ has an upward curvature. A straight line would not give a very good fit. Therefore, we fitted a quadratic to the data and then subtracted 1.0 from the constant term of the quadratic in order that the fit might lie slightly under all the data points (except for two which are obviously out-of-line). The function so obtained is:

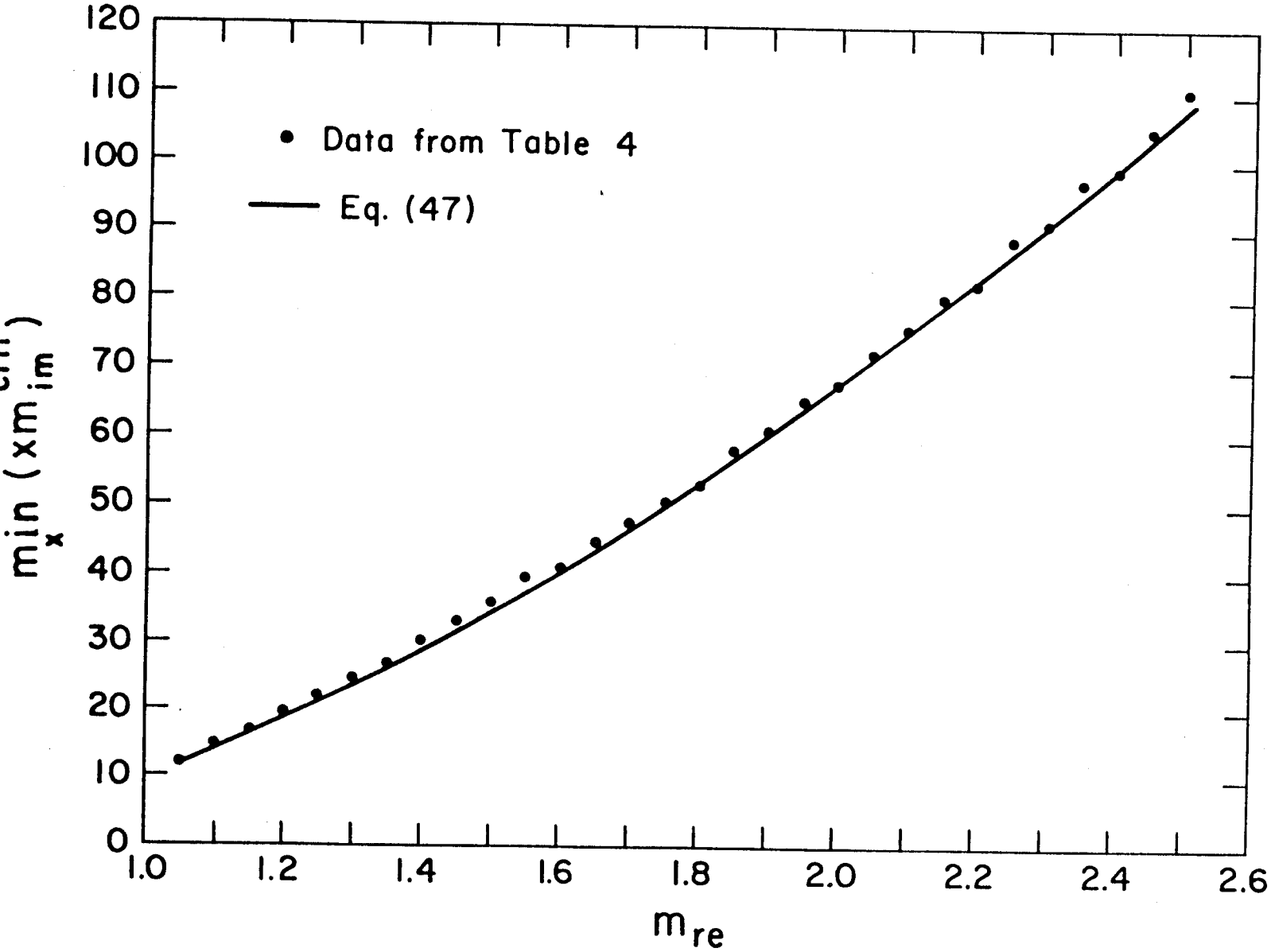

120
110
100
90
80
70
60
50
40
30
20
10
0
1.0
1.2
1.4
1.6
1.8
2.0
2.2
2.4
2.6
• Data from Table 4
Eq. (47)
$\min_x (x m_{im}^{crit})$
m_{re}

Figure 1

Table 4. The smallest value of the product xm_{im}^{crit} at each value of m_{re}, where m_{im}^{crit} is the value of m_{im} above which up-recurrence on A_n fails ($x \leq 10{,}000$ for this study).

m_{re}	$\min_x(xm_{im}^{crit})$	m_{re}	$\min_x(xm_{im}^{crit})$
1.05	12.2	1.80	53.0
1.10	14.9	1.85	57.8
1.15	16.9	1.90	60.7
1.20	19.6	1.95	65.2
1.25	22.2	2.00	67.5
1.30	24.7	2.05	72.1
1.35	27.0	2.10	75.5
1.40	30.3	2.15	80.0
1.45	33.2	2.20	81.9
1.50	36.0	2.25	88.6
1.55	39.5	2.30	91.2
1.60	40.7	2.35	97.2
1.65	44.5	2.40	99.0
1.70	47.4	2.45	104.8
1.75	50.6	2.50	110.4

$$f(m_{re}) = 16.35\ m_{re}^2 + 8.42\ m_{re} - 15.04 \tag{47}$$

and it is plotted in Fig. 1 as a solid line.

From Eq. (46) we deduce that up-recurrence may be used for A_n if

$$m_{im} < \frac{f(m_{re})}{x} \tag{48}$$

but that, otherwise, down-recurrence should be used. This criterion, since it concerns the onset of numerical instability, is obviously precision-dependent, although probably not sensitively so. Also, being only empirical, it may break down for values of m_{re} which are much larger than 2.50, as may occur for water in the microwave region for example.

6. NUMBER OF TERMS IN MIE SERIES

Dave (1969a) stopped summing the Mie series at the first value of n for which

$$|a_n|^2 + |b_n|^2 < 10^{-14} \tag{49}$$

(cf. Eq. 16). He was confident that this led to 6 significant digits in the results. We used this criterion for several years, but were forced to abandon it in order to attain vector speed in our Mie codes.

The reason is simply that vector processing is not possible for a loop containing a test like (49). Calculation of the entire vectors $\{a_1,\ldots\}$ and $\{b_1,\ldots\}$ at one stroke cannot occur if the value of each a_n,b_n decides whether or not to calculate a_{n+1}, b_{n+1}.

We were just as glad to abandon Dave's criterion for two other reasons as well. First, the figure 10^{-14} may not be applicable to computers of differing precision. Related to this is the fact that the upward recurrence on ψ_n, whose fall to zero for $n \gtrsim x$ is the primary cause of Mie series convergence, rapidly becomes unstable near the very point where (49) must be satisfied. Eq. (49), in effect, is asked to "head off" the mushrooming instability in the ψ_n calculation, and this seems too heavy a burden to lay upon it.

Second, even though our codes always carried the same precision (14 digits) as Dave's, the test (49) would sporadically fail in the middle of lengthy calculations, typically for $x > 800$ or so. Examination of every such case revealed that failure was caused by $|a_n|^2 + |b_n|^2$ falling to, but not below, some low level (e.g., 5×10^{-14}). Further examination showed that convergence was nevertheless

satisfactory in such cases, and the test (49) merely failed to recognize it.

For these reasons, the test (49) was replaced by an a priori estimate of N, the number of terms in the various Mie series (6-10). We first determined N for a wide range of size parameters ($0.1 \leq x \leq 20{,}000$) and refractive indices ($1.05 \leq m_{re} \leq 2.50$; $0 \leq m_{im} \leq 1$) using criterion (49) with 5×10^{-14} replacing 10^{-14}. Table 5 shows selected results for the range $[N_{min}, N_{max}]$ of values of N at each size parameter.

It is immediately apparent that N exhibits only a slight dependence on refractive index. This bears out what we said earlier about ψ_n and χ_n, which are functions of x alone, being the prime controllers of convergence. And since ψ_n decreases rapidly for $n \gtrsim x$, it is clear why $N \sim x$ is a good first approximation.

Several authors (e.g., Khare, 1976) have suggested, on theoretical grounds, the following functional form for N:

$$N = x + cx^{1/3} .$$

We have found that the following modification of this formula fits our data for N_{max} exceptionally well:

$$N_{max} = \begin{cases} x + 4x^{1/3} + 1 & 0.02 \leq x \leq 8 \\ x + 4.05x^{1/3} + 2 & 8 < x < 4200 \\ x + 4x^{1/3} + 2 & 4200 \leq x \leq 20{,}000 \end{cases} . \qquad (50)$$

This gives an almost perfect fit to 135 N_{max} values which we generated. It is low by 1 at 8 of these data points, and high by 1 or (rarely) 2 at 35 of them; at the remainder, it is exact. We set $N = N_{max}$.

Table 5. The range $[N_{min}, N_{max}]$ of the number of terms in the Mie series, as a function of size parameter x. The range was determined by varying Re(m) from 1.05 to 2.50 and Im(m) from 0 to 1 in small steps.

x	$N_{min} - N_{max}$	x	$N_{min} - N_{max}$
0.1	2-3	3,000	3,052-3,058
0.3	3-3	6,000	6,068-6,074
1	5-5	8,000	8,076-8,080
3	8-9	10,000	10,081-10,087
10	18-20	12,000	12,086-12,092
33	44-47	14,000	14,090-14,097
100	117-120	16,000	16,087-16,101
333	357-362	18,000	18,094-18,105
1,000	1,038-1,041	20,000	20,102-20,108

7. MIE SCATTERING SUBROUTINES

The present section describes the two Mie scattering subroutines MIEV0 (Sec. 7.2) and MIEV1 (Sec. 7.3), which are based on the techniques and equations given in Secs. 3-6. Why two subroutines are necessary was explained in Sec. 2. These subroutines are listed and flowcharted in Appendices I and II. Detailed timing studies are presented in Sec. 7.5 and a description of testing procedures in Sec. 7.4. Sample results from the subroutines are given in Appendix III.

An important a priori decision was whether or not to take advantage of FORTRAN complex arithmetic. In my earlier Mie routines, I avoided complex arithmetic entirely because it tended to be considerably slower than real arithmetic (for example, multiplying a real by a complex variable would compile as four floating point multiplies, instead of two). But the loss of code readability and simplicity from separating every expression into real and imaginary parts was drastic. In recent years, the dramatic rise in use of complex arithmetic, primarily because of the invention of the Fast Fourier Transform, has forced compiler-writers to do a much better job with it. It is, therefore, increasingly possible to write dramatically simpler Mie codes without loss of speed, using complex arithmetic; that is the path chosen here.

Double precision arithmetic is not used anywhere in the subroutines, but users whose machines carry less than roughly 11 digits in single precision should carry out the ζ_n and A_n recurrences in double precision and truncate the results back to single precision.

The two subroutines share a substantial amount of coding, in particular: the small-x limit; the N formula; the A_n computation; and the θ-loops where $S^{\pm}$, S_1 and S_2 are calculated. They both rely on compiler recognition of repeated subexpressions (an increasingly common capability).

The input and output variables used by MIEV0 and MIEV1 are in a common block (an argument list would be computationally slower) and are thoroughly described in comment cards. The only question which might arise is: How would one go about picking a value of the input variable N2CUT? (The code assumes Im(m) = 0, and takes faster branches, when $|\mathrm{Im}(m)| \leq$ N2CUT.) There is no easy answer. It depends on the largest size of particle being considered; the largest optical depth being considered (individual sphere absorptions may be very small and yet add up to a substantial value in an optically thick medium); and how small absorption must be before one is willing to neglect it. The user must either experiment a little with N2CUT--or set it to zero and forget about it.

7.1 VECTORIZATION

A few preliminary comments on vector processing are in order (Johnson, 1978, gives a good introduction to the subject in relation to the CRAY-1 computer). It should be emphasized at the outset that the Mie codes are designed to execute perfectly well, albeit more slowly, on ordinary "scalar" computers; but they have been written specifically to take advantage of vector processing when it is available.

Vector processing is applicable to loops of a very special form, in which entire arrays can be operated on as if they were scalars. Any so-called "vector dependencies" (e.g., in the recurrences for $\{\zeta_0,\ldots,\zeta_N\}$ or $\{A_1,\ldots,A_N\}$), where one array element depends on one or two previous array elements, prevents "vectorization" of the associated loop. Such recurrences are unvectorizable in principle.

Other constructs, such as scalar temporary variables, user-defined function or subroutine references, and certain IF statements, may superficially prevent vectorization of a loop in which they occur; but a redesign of the loop coding often allows vectorization. Loops in which only some parts are unvectorizable can often be split into two loops, the first of which isolates the unvectorizable parts and the second of which is fully vectorizable. Some operations, like summing the elements of an array, which at first glance seem unvectorizable, can be "partially vectorized"; this is exactly what is done in MIEV1 to sum the series for Q_{ext}, Q_{sca} and g. These tricks, and others, are discussed at more length by Higbie (1978). All this, of course, leads to code which seems, in some sense, "unnatural," but one is quickly reconciled to such unnatural code when one sees the dramatic speed increases it produces.

In Mie calculations there are two types of loops: over number N of terms in the Mie series and over number N_{ang} of angles. A third type of loop, over size parameter x, is often used, for example, when integrating Mie quantities over a size distribution. This loop is normally outside the Mie subroutine. Some advantage could be reaped by having this loop inside the Mie subroutine. In particular,

unvectorizable elements in MIEV0 and MIEV1, like the A_n and ζ_n recurrences, could be vectorized by making the x-loop the innermost one and adding an extra x-dimension to A_n, ζ_n, etc. However, two disadvantages also accrue: the Mie subroutine would be much less general-purpose, and the memory requirements could become prohibitive. Users who wish to try this route should be warned that: (a) if N_{ang} is, say, 30 or more, little speed-up will be realized, because the N_{ang}-loop dominates the execution time; (b) N depends on x, and with x-loops innermost and N-loops outside, this dependence cannot be accounted for (taking N sufficient for the largest x could cause the A_n and ζ_n up-recurrences for smaller x's to go wildly unstable).

Calculating S_1 and S_2 involves one N-loop and one N_{ang}-loop, nested. But only inner loops are vectorizable. In the N-loop, S_1 and S_2 are computed by summing, which is only a partially vectorizable operation; hence, the natural choice was to make the N_{ang}-loop the inner one (it is completely vectorizable, provided several arrays are furnished to contain temporary variables). The N_{ang}-loop-innermost structure is found in both MIEV0 and MIEV1.

7.2 MIEV0

MIEV0 aims at minimum storage. The only significant dimensioned array is for A_n. (Arrays dimensioned only by the number of angles are usually short and rarely contribute significantly to the storage burden.) It requires a storage allocation because it may have to be precomputed by down-recurrence. The rest of the computation unfolds naturally as an up-recurrence requiring no storage of intermediate

results. Because of this, MIEV0 appears simple and straightforward, especially by comparison to MIEV1.

The only significant vectorization in MIEV0 is of the N_{ang}-loops necessary in the S_1,S_2 computation, as described in Sec. 7.1. But, as the timing studies will show (Sec. 7.5), these are by far the most important loops to vectorize if S_1 and S_2 are computed at any reasonable number of angles.

By setting $N_{ang} = 0$ in MIEV0, one can calculate Q_{ext}, Q_{sca} and g only, which is all one needs for simple radiative transfer approximations like delta-Eddington (Wiscombe, 1977). But MIEV0 is two to three times slower than MIEV1 in this mode, so MIEV1 should be used for such applications whenever possible.

The numerical coefficients needed in the various Mie series are formed in MIEV0 by the following efficient procedure:

$$2n + 1 = n + (n + 1) \tag{51a}$$

$$\frac{(n + 1)(n - 1)}{n} = n - \frac{1}{n} \tag{51b}$$

$$\frac{n + 1}{n} = 1 + \frac{1}{n} \tag{51c}$$

$$\frac{2n + 1}{n(n + 1)} = \frac{1}{n} + \frac{1}{n + 1} \quad . \tag{51d}$$

Only one division is actually done (1/n+1) because the value of 1/n is saved from the previous pass through the n-loop. The coefficient $(-1)^{n+1}$ is contained in an integer variable which is initialized to 1 and then has its sign flipped each time through the n-loop.

7.3 MIEV1

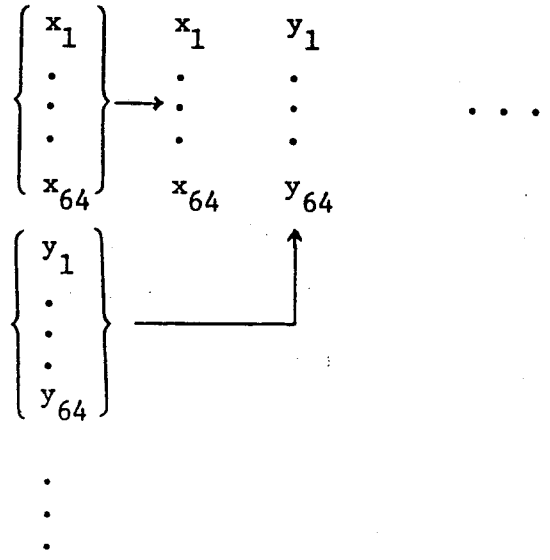
MIEV1 aims for the maximum amount of vectorization. It vectorizes all those N_{ang}-loops which are vectorized in MIEV0 and, in addition, it vectorizes several N-loops. To do so it divides computations into separate loops which were combined into a single loop in MIEV0. Also, MIEV1 requires 11 arrays of length N, rather than 2 as in MIEV0 (a complex array counting as 2), even though every effort has been made to minimize storage.

Four of the 11 N-arrays in MIEV1 contain the numerical coefficients

$$n,\ 2n+1,\ \frac{2n+1}{n(n+1)},\ \text{and}\ \frac{n(n+2)}{n+1} \tag{52}$$

It is necessary to store these in arrays in order for certain N-loops to vectorize. Vectorization is inhibited unless all operands in an arithmetic expression are either constants or vectors; functions of the loop index like (52) are, as they stand, neither.

Summing of the series for Q_{ext}, Q_{sca} and g (Eqs. 6-8) is done in MIEV1 using a partially-vectorized subroutine. It is based on the following conceptual rearrangement into matrix form of the linear array being summed:



Each successive group of 64 elements in the array is regarded as a column of the matrix. Matrix row sums are then formed in a loop which vectorizes:

$$S_i = x_i + y_i + \ldots \qquad (i = 1 \text{ to } 64) \quad . \tag{53}$$

Then these row sums are added in an unvectorizable loop to give the desired sum:

$$S = S_1 + \ldots + S_{64} \quad . \tag{54}$$

It is because this last loop is unvectorizable that we call the procedure only "partially vectorized."

The above procedure has been coded into the FORTRAN subroutine TOTAL, listed along with MIEV1 in Appendix II. Obviously, TOTAL offers no speed advantage when less than 65 elements are to be summed; in that case TOTAL branches to a simple scalar loop

```
      TOTAL = A(1)
      DO 1 I = 2,N                                    (55)
    1 TOTAL = TOTAL + A(I)  .
```

According to Johnson (1978), summing may be speeded up even more on the CRAY-1 by doing a vector add between steps (53) and (54). This extra step is vectorizable and results in having to sum only 8 elements, instead of 64, in the final unvectorizable step (54). Unfortunately, it is necessary to use assembly language coding to reap this extra speed and this would make MIEV1 nonportable. But those who use MIEV1 on the CRAY-1 may want to avail themselves of this fast sum routine.

Timing comparisons for sums of various lengths are presented in Table 6 for the three summing methods: standard (Eq. 55), TOTAL, and assembly language. For sums of more than 40 terms, assembly language summing takes only one-third to one-sixth the time of TOTAL. Note also that TOTAL is actually slower than the standard method for sums of less than about 100 terms; the overhead associated with calling a subroutine outweighs the benefits of vectorization up to this point. But beyond 100 terms, TOTAL quickly shows its advantage on just those sums which take the most computer time. For sums of 1000 terms and more, it takes only one-fourth to one-fifth the computer time of the standard method, albeit a factor of 3 to 4 more time than the assembly language method.

7.4 TESTING

The following parts of the basic routine MIEV0 were tested extensively by themselves:

- the recurrences for ψ_n, χ_n and A_n, particularly their possible instability (Sec. 3)
- the Lentz method (Sec. 3)
- the recurrences for π_n, τ_n (Sec. 3)
- the small-x limit (Sec. 4)
- the empirical function N(x) (Sec. 6)
- the A_n up-recurrence criterion (Sec. 5) .

MIEV0 was then tested as a whole for internal consistency, for stability and for well-conditioning; its results were compared to a considerable variety of published Mie scattering data, and to the author's previous Mie code.

Table 6. Timing comparisons between various summing methods: standard (Eq. 55), partially-vectorized FORTRAN (TOTAL), and assembly language (Johnson, 1978).

Number of terms summed	Standard method time (millisec)	Ratio of TOTAL to standard method time	Ratio of assembly language to standard method time
5	2.1×10^{-3}	1.8	1.3
10	3.7×10^{-3}	1.5	0.77
20	7.0×10^{-3}	1.3	0.45
40	1.3×10^{-2}	1.2	0.27
60	2.0×10^{-2}	1.2	0.21
100	3.3×10^{-2}	0.93	0.15
150	4.9×10^{-2}	0.68	0.12
200	6.6×10^{-2}	0.56	0.10
300	9.8×10^{-2}	0.43	0.084
500	0.16	0.32	0.071
1,000	0.33	0.25	0.061
2,000	0.65	0.21	0.057
20,000	6.5	0.18	0.052

The internal consistency checks consisted in ascertaining, in numerous test cases, that the code results were the same to at least 6 significant digits when (a) various formulas were restructured, (b) varying levels of precision were used, and (c) convergence flags were varied within reasonable limits.

The stability and well-conditioning tests consisted in exercising the code over broad ranges of x (up to 20,000) and m (real part from 1.05 to 2.50; imaginary part from 10^{-7} to 1) and seeing if any overflows or unreasonble results turned up. No published data exist over most of these ranges, but there are a number of checks which one can make; e.g., that $Q_{ext} \rightarrow 2$ and Q_{sca} and g approach well-known asymptotes as $x \rightarrow \infty$; that these approaches should be more rapid--the larger Im(m) is; that increasing Im(m) damps out the ripple structure--the more so the larger x is; that rainbows and glories should occur; and so on. Numerous graphs were made of the results and, after considerable experience with Mie scattering data, I have found that calculational errors are relatively easy to spot on such graphs since they cause deviations from what is usually a fairly regular pattern. The final form of MIEV0 in Appendix I has passed all such tests.

The published data against which MIEV0 was tested included both graphs and tables in Van de Hulst (1957), Irvine (1965), Kerker (1969), Dave (1968b), and Deirmendjian (1969); and tables in Deirmendjian (1963), Denman et al. (1966), and Dave (1968a). Such data are restricted to $x \leq 1000$ and, while being able to reproduce them gave considerable confidence in the codes, the sorts of tests described in the last paragraph are equally as important; for extrapolating the code to $x \leq 20{,}000$ might overwhelm numerical techniques which are perfectly adequate for $x \leq 1000$.

Of course the Mie codes were modified and improved several times after their initial testing. Rather than rerun the tests, comparisons were made for wide ranges of x and m between all output quantities of the new and old versions. This enabled errors to be quickly detected and ensured that the final versions presented here passed all the tests that their predecessors did. It also was the method used to test MIEV1 (against MIEV0).

Sample results from MIEV1, rounded to 6 significant digits, are presented in Appendix III. Users should reproduce these results before attempting to apply MIEV0 or MIEV1. There are 8 cases: x = 10, 100, 1000, and 5000; for each x, m = 1.5 and m = 1.5 - 0.1 i. Q_{ext}, Q_{sca}, $Q_{abs} = Q_{ext} - Q_{sca}$, g, and the time to execute each case are shown, as well as S_1 and S_2 for $\theta = 0°$ (5°) 180°. The intensity and degree of polarization (Eqs. 44-45) are also shown; these were tabulated by Dave (1968a) for our two x = 1000 cases, and the Appendix III results agree with his to all digits which he tabulates, except that the sign of our degree of polarization is opposite to his (we believe his is wrong in this respect).

Note that Dave gives execution times of 50.36 sec. (m = 1.5) and 52.32 sec. (m = 1.5-0.1 i) for his test cases. The MIEV1 times are 3300 and 2900 times faster, respectively.

7.5 TIMING

Claims of computer code timing superiority turn up frequently, although rarely are they backed up by the sort of thorough, quantitative study presented below for MIEV0 and MIEV1. Therefore, I would like to begin with a few general comments on the meaning and utility (or lack thereof) of timing comparisons.

My main observation--and this has been seasoned by years of experience with a variety of codes and computing systems--is that many claims of timing superiority are meaningless. They are simply down in the noise level. I base this incredulity on several elements of experience.

First, the comparisons are almost never thorough or comprehensive, but rather are based on a tiny sample of cases which may not be representative or even important.

Second, timing on a computer is inherently a noisy operation, and the results may not be exactly reproducible. It depends on the workload, the I/O burden, and many other unreckonable factors. It may vary even among computers of the same kind, depending on the sophistication of their operating systems and compilers (which are often a local product or an extensively modified company product). The 'perfect' timing study would be a pure CPU job, involving no I/O other than printing the times at the end, and executing with no other jobs present. These conditions are rarely met.

Third, the putatively slower code is not usually optimized. I have seen professional 'code speeder-uppers' reorganize a code, put the slowest or most frequently executed parts in assembly language, etc., and improve the timing by several factors. Indeed, after such optimization, the slower code might become the faster one.

As a result, I have come to regard, as a rough rule-of-thumb, that only a timing factor of 4 or more between two entirely different codes is significant. (Of course, timing improvements from modifying similar codes on the same computer are significant at a much lower level, possibly as low as 5-10%.)

Another point to bear in mind is what benefit will accrue from a timing improvement. A code executing in 1 microsecond will not benefit from a speed-up unless it is called many millions of times. At the other end of the scale, a code requiring 100,000 hours will not become any more feasible with a factor of 10 speed-up. It is usually only calculations requiring times in the mid-range (seconds to hours) which will benefit from a speed-up. Mie calculations are of such a type.

By way of an example of a useful timing comparison, Table 7 shows times for Dave's (1969a) Mie code, for 182 angles and x = 1, 10, 100, 1000 and 5000; and corresponding times for vectorized MIEV1. MIEV1 is 3000-4000 times faster, which divides up as follows. Our scalar-mode CRAY-1 is roughly 100 times faster than Dave's IBM 360/50. Another factor of roughly 5 is due to our improvements in numerical technique and a final factor of about 7 can be ascribed to our partially-vectorized code design.

Palmer (1977), using the same kind of computer as Dave, reported a new numerical technique (continued fractions instead of series) which gave factors of 9-15 improvement over Dave's times in Table 7 for x = 1, 10 and 100. The fact that both Palmer and I obtained such large improvements from entirely different directions suggests that, from the timing standpoint, Dave's codes were far from ideal.

We now examine the times required by three versions of our codes: unvectorized MIEV0, vectorized MIEV0, and vectorized MIEV1. These times, in CRAY-1 milliseconds, are presented in Tables 8-10, respectively, for size parameters from 1 to 5000, for numbers of angles from 0 to 255, and for two imaginary refractive indices--0 and 0.1. Each

Table 7. Execution times for Dave (1969a) Mie code compared to vectorized MIEV1 time. 182 angles were used for both codes. Dave used an IBM 360/50, which is roughly 100 times slower than the CRAY-1 in its normal scalar mode.

x	Dave (1969a) code time (sec)	vectorized MIEV1 code time (sec)	ratio of times
0.1	0.7	1.8×10^{-4}	3900
1	1.1	3.6×10^{-4}	3100
10	3.7	9.8×10^{-4}	3800
100	22	5.4×10^{-3}	4100
1000	194	4.56×10^{-2}	4300
5000	945	0.222	4300

Table 8. CRAY times in milliseconds to execute the unvectorized MIEV0 code for various combinations of size parameter and number of angles. Each time represents an average over Re(m) = 1.1 (0.2) 2.5.

Im(m) = 0

No. of Angles	Mie Size Parameter							
	1	3.3	10	33	100	333	1000	5000
0	0.081	0.11	0.19	0.41	1.0	3.0	8.4	41
3	0.13	0.18	0.33	0.72	1.8	5.3	15	74
7	0.16	0.24	0.44	0.98	2.4	7.3	21	101
15	0.23	0.35	0.66	1.5	3.7	11	32	155
31	0.38	0.58	1.1	2.5	6.3	19	54	262
63	0.66	1.0	2.0	4.5	11	34	98	476
127	1.2	2.0	3.7	8.6	22	65	186	905
255	2.4	3.8	7.3	17	42	127	363	1760

Im(m) = 0.1

0	0.099	0.14	0.24	0.50	1.2	3.7	12	53
3	0.14	0.21	0.37	0.82	2.0	6.1	19	86
7	0.18	0.27	0.48	1.1	2.7	8.0	25	113
15	0.25	0.38	0.70	1.6	3.9	12	36	167
31	0.39	0.61	1.1	2.6	6.5	20	58	274
63	0.68	1.1	2.0	4.6	12	35	102	488
127	1.3	2.0	3.8	8.7	22	66	190	918
255	2.4	3.8	7.3	17	42	127	367	1780

Table 9. CRAY times in milliseconds to execute the vectorized MIEV0 code for various combinations of size parameter and number of angles. Each time represents an average over Re(m) = 1.1 (0.2) 2.5.

Im(m) = 0

No. of Angles	Mie Size Parameter							
	1	3.3	10	33	100	333	1000	5000
0	0.081	0.11	0.19	0.41	1.0	3.0	8.4	41
3	0.12	0.18	0.32	0.70	1.7	5.1	15	71
7	0.13	0.18	0.32	0.71	1.7	5.2	15	72
15	0.14	0.20	0.35	0.78	1.9	5.7	16	79
31	0.16	0.23	0.41	0.92	2.3	6.8	19	94
63	0.20	0.30	0.54	1.2	3.0	8.9	25	123
127	0.28	0.43	0.80	1.8	4.4	13	38	182
255	0.46	0.72	1.4	3.0	7.6	23	66	315

Im(m) = 0.1

0	0.099	0.14	0.24	0.50	1.2	3.8	12	53
3	0.14	0.20	0.36	0.78	1.9	5.9	18	83
7	0.14	0.21	0.37	0.80	2.0	6.0	19	84
15	0.15	0.22	0.40	0.87	2.1	6.5	20	91
31	0.17	0.25	0.46	1.0	2.5	7.5	23	106
63	0.21	0.32	0.58	1.3	3.2	9.6	29	135
127	0.30	0.45	0.82	1.8	4.6	14	41	194
255	0.48	0.74	1.4	3.1	7.8	23	69	327

Table 10. CRAY times in milliseconds to execute the vectorized MIEV1 code (the fastest one) for various combinations of size parameter and number of angles. Each time represents an average over Re(m) = 1.1 (0.2) 2.5.

Im(m) = 0

No. of Angles	Mie Size Parameter							
	1	3.3	10	33	100	333	1000	5000
0	0.062	0.074	0.11	0.19	0.38	0.99	2.7	13
3	0.11	0.15	0.24	0.50	1.2	3.3	9.4	45
7	0.11	0.15	0.25	0.51	1.2	3.5	9.7	47
15	0.12	0.17	0.28	0.58	1.4	4.0	11	54
31	0.14	0.20	0.34	0.72	1.7	5.0	14	69
63	0.18	0.26	0.46	1.0	2.4	7.1	20	99
127	0.27	0.39	0.71	1.6	3.8	11	32	157
255	0.45	0.68	1.3	2.8	7.0	21	60	290

Im(m) = 0.1

0	0.083	0.10	0.14	0.25	0.53	1.5	5.6	21
3	0.13	0.18	0.28	0.57	1.3	3.8	12	53
7	0.13	0.18	0.29	0.59	1.4	4.0	13	55
15	0.14	0.20	0.32	0.66	1.5	4.5	14	62
31	0.16	0.23	0.39	0.80	1.9	5.5	17	77
63	0.20	0.29	0.51	1.1	2.6	7.7	23	106
127	0.29	0.42	0.75	1.6	4.0	12	35	165
255	0.47	0.73	1.3	2.9	7.2	22	63	298

time represents an average over eight real refractive indices from 1.1 to 2.5 in steps of 0.2.

For Im(m) = 0, up-recurrence is always used for A_n, as well as faster code branches for calculating a_n,b_n, so these times are invariably less (typically by 5-30%) than the corresponding ones for Im(m) = 0.1. For Im(m) = 0.1, up-recurrence for A_n may or may not be used, depending on Eq. (48); for Re(m) = 1.1, for example, it is used only for the $x \leq 100$ cases, while for Re(m) = 2.5 it is used for all except the x = 5000 case. The sudden switch to down-recurrence on A_n thus occurs somewhere between the x = 100 and x = 5000 columns of the tables, but the elevating effect of this on the quoted times is diminished somewhat by averaging over the eight values of Re(m).

The patterns in Table 8 are typical of what one may expect on computers without vector capabilities. For fixed x, the N_{ang}-loop embedded in the N-loop increasingly dominates the computation time, until, for $N_{ang} \geq 31$, the time is almost linear in N_{ang}. For fixed N_{ang}, the time rises a bit less than linearly with N or, equivalently, with $x + 4x^{1/3}$ (cf. Eq. 50).

For vectorized MIEV0, Table 9, the times for 0 and 3 angles are almost identical to those for unvectorized MIEV0, Table 8; this is to be expected, since only N_{ang}-loops have been vectorized. But for $N_{ang} > 3$, the times in Table 9 increase much more slowly with rising N_{ang} than they did in Table 8. For 15 angles, there is an advantage of 1.8-1.9 over unvectorized MIEV0 for all x; for 31 angles, 2.4-2.8; for 63 angles, 3.5-3.9; up to 5.2-5.7 for 255 angles. This advantage factor rises roughly logarithmically in N_{ang}, increasing by about unity

every time N_{ang} doubles. As x increases for fixed N_{ang}, the times in Table 9 rise in almost the exact same way they did in Table 8 (linearly in N); again, this is not surprising since no N-loops are vectorized.

The fastest code times are those for vectorized MIEV1, in Table 10. Compared to Table 9, improvement over MIEV0 is greatest for 0 angles--as large as a factor of 3 and at least a factor of 2 whenever $x \geq 33$. This shows the advantage from vectorizing N-loops in its best light. For $N_{ang} \geq 3$, on the other hand, vectorized MIEV0 never takes over 60% more time than vectorized MIEV1; vectorized MIEV1 offers the most advantage for $3 \leq N_{ang} \leq 63$ and $x \leq 100$, in which regime vectorized MIEV0 takes 25-60% more time. (Of course, it is just this large-x regime which requires the most computer time.) But for 255 angles, Table 9 times are never more than 10% above those in Table 10, reflecting the almost total dominance of the N_{ang}-loops at that point.

There is a big rise, by a factor of 2-3, between the times for $N_{ang} = 0$ and those for $N_{ang} = 3$ in Table 10. But after this initial jump, the further rise as N_{ang} increases is slow; it is necessary to go all the way to 63 angles to double the $N_{ang} = 3$ time. Beyond 63 angles, the rise in execution time is more rapid, approaching linearity in N_{ang}; this reflects the vector length of 64 on the CRAY-1.

For $x \leq 33$, the Table 10 times rise considerably more slowly than linearly in N, reflecting the vectorization of N-loops. For $x \geq 100$, the times go up more nearly linearly in N, again due to the CRAY-1 vector length.

Using the assembly language sum routine (Sec. 7.3) can speed-up vectorized MIEV1 by anywhere from 1% to 20%. Unfortunately, the larger

speed gains are associated with the smaller values of x and N_{ang}, which take less computer time anyway. For example, for x = 33, the gain ranges from 15-18% for 0 angles to 0-2% for 255 angles; while for x = 333 it never exceeds 7%. For maximum speed the fast sum routine can certainly be recommended, especially for $N_{ang} = 0$ cases, but no dramatic improvements in timing result therefrom.

These timing studies furnish a more solid basis than has heretofore existed for (a) estimating how much time a particular Mie computation will require and (b) establishing by how much, and in what cases, future Mie codes improve on these times.

8. SUMMARY

This document describes a number of improvements in numerical technique for Mie scattering calculations, and incorporates them into two well-documented and tested computer codes. These improvements are as follows:

1. Design for vector processing (Sec. 7.1)
2. Lentz method for starting the downward recurrence of A_n (Eqs. 23-35)
3. New criterion for down-recurrence of A_n (Sec. 5)
4. Better treatment of small-particle limit (Sec. 4)
5. A priori formulas for number of series terms (Sec. 6)
6. Simpler $\pi_n(\mu)$, $\tau_n(\mu)$ recurrences (Eqs. 37, 38)
7. Faster branches for no-absorption cases (Sec. 7.5)
8. $S^{\pm}$, rather than S_1, S_2, calculated internally, for greater speed (Eqs. 9-12)
9. Complex arithmetic, no double precision (Secs. 3, 7) .

The codes execute some 30 to 40 times faster than those of Dave (1968a, 1969a) even after differences in machine speed are factored out. While this may seem a bit like flogging a dead horse, considering that Dave's codes may have been very inefficient to begin with (see Sec. 7.5), the statistic is important because many people are still using Dave's codes or slight variations thereof.

With codes such as those presented herein, Mie calculations which were literally unthinkable only 5 to 10 years ago may now be done routinely.

REFERENCES

Abramowitz, M., and I. Stegun, Eds., 1965: Handbook of Mathematical Functions. Dover Press, New York, 1046 pp.

Born, M., and E. Wolf, 1975: Principles of Optics. Pergamon Press, New York (5th edition).

Chýlek, P., J. T. Kiehl and M. K. W. Ko, 1978: Narrow resonance structure in the Mie scattering characteristics. Appl. Opt., 17, 3019-3021.

Dave, J. V., 1968a: Subroutines for computing the parameters of the electromagnetic radiation scattered by a sphere. Report No. 320-3237, IBM Scientific Center, Palo Alto, CA, 65 pp.

Dave, J. V., 1968b: Scattering of visible light by large water spheres. Appl. Opt., 8, 155-164.

Dave, J. V., 1969a: Scattering of electromagnetic radiation by a large, absorbing sphere. IBM J. Res. Dev., 13, no. 3, 302-313.

Dave, J. V., 1969b: Effect of coarseness of the integration increment on the calculation of radiation scattered by polydispersed aerosols. Appl. Opt., 8, 1161-1167.

Deirmendjian, D., 1963: Tables of Mie Scattering Cross Sections and Amplitudes. RAND Report R-407-PR, Rand Corporation, Santa Monica, CA.

Deirmendjian, D., 1969: Electromagnetic Scattering on Spherical Polydispersions. American Elsevier, New York, NY.

Denman, H., W. Heller and W. Pangonis, 1966: Angular Scattering Functions for Spheres. Wayne State University Press, Detroit, MI.

Higbie, Lee, 1978: Speeding Up FORTRAN (CFT) Programs on the CRAY-1. CRAY Research Technical Note 2240207 (available from CRAY Research, 1440 Northland Drive, Mendota Heights, MN 55120).

Irvine, W. M., 1965: Light scattering by spherical particles: Radiation pressure, asymmetry factor, and extinction cross-section. J. Opt. Soc. Amer., 55, 16-21.

Johnson, P. M., 1978: An introduction to vector processing. Computer Design, February.

Kattawar, G. W., and G. N. Plass, 1967: Electromagnetic scattering from absorbing spheres. Appl. Opt., 6, 1377-1382.

Kerker, M., 1969: The Scattering of Light and Other Electromagnetic Radiation. Academic Press, New York, NY.

Kerker, M., P. Scheiner and D. D. Cooke, 1978: The range of validity of the Rayleigh and Thomson limits for Lorenz-Mie scattering. J. Opt. Soc. Amer., 68, 135-137.

Khare, V., 1976: Short-Wavelength Scattering of Electromagnetic Waves by a Homogeneous Dielectric Sphere. Ph.D. Thesis, University of Rochester, NY (available from University Microfilms, Ann Arbor, MI).

Lentz, W. J., 1976: Generating Bessel functions in Mie scattering calculations using continued fractions. Appl. Opt., 15, 668-671.

Logan, N. A., 1965: Survey of some early studies of the scattering of plane waves by a sphere. Proc. IEEE, 53, 773-785.

Palmer, T. Y., 1977: Mie scatter from continued fractions. In Advances in Laser Technology for the Atmospheric Sciences. J. Trolinger and W. Moore, Eds., Soc. Photo-Optical Inst. Eng., Bellingham, WA, pp. 121-124.

Rosasco, G. J., and H. S. Bennett, 1978: Internal field resonance structure: Implications for optical absorption and scattering by microscopic particles. J. Opt. Soc. Am., 68, 1242.

Shifrin, K. S., 1951: Scattering of Light in a Turbid Medium. English translation, NASA TTF-477, Washington, D.C., 1968.

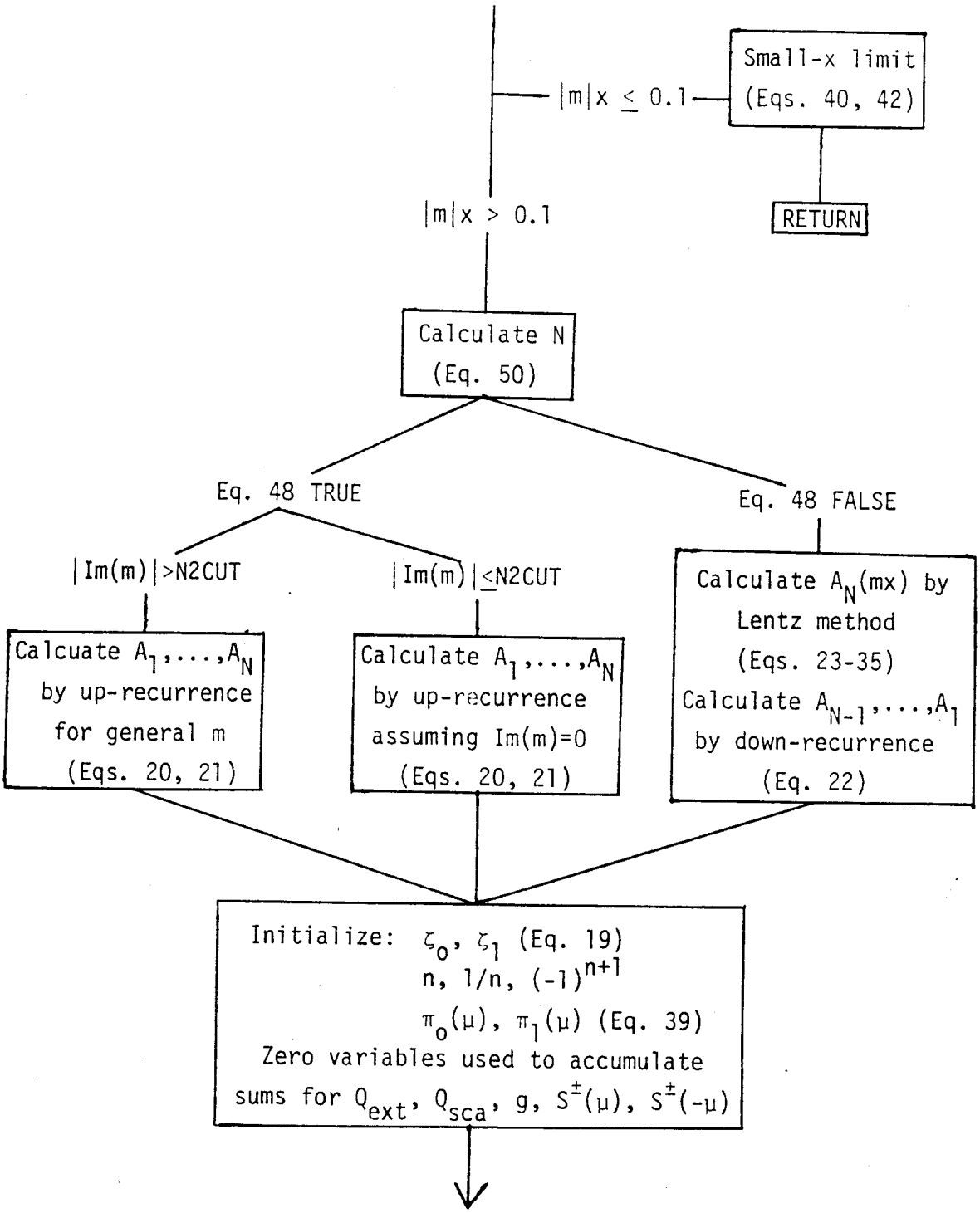
Van de Hulst, H. C., 1957: Light Scattering by Small Particles. Wiley & Sons, New York, NY.

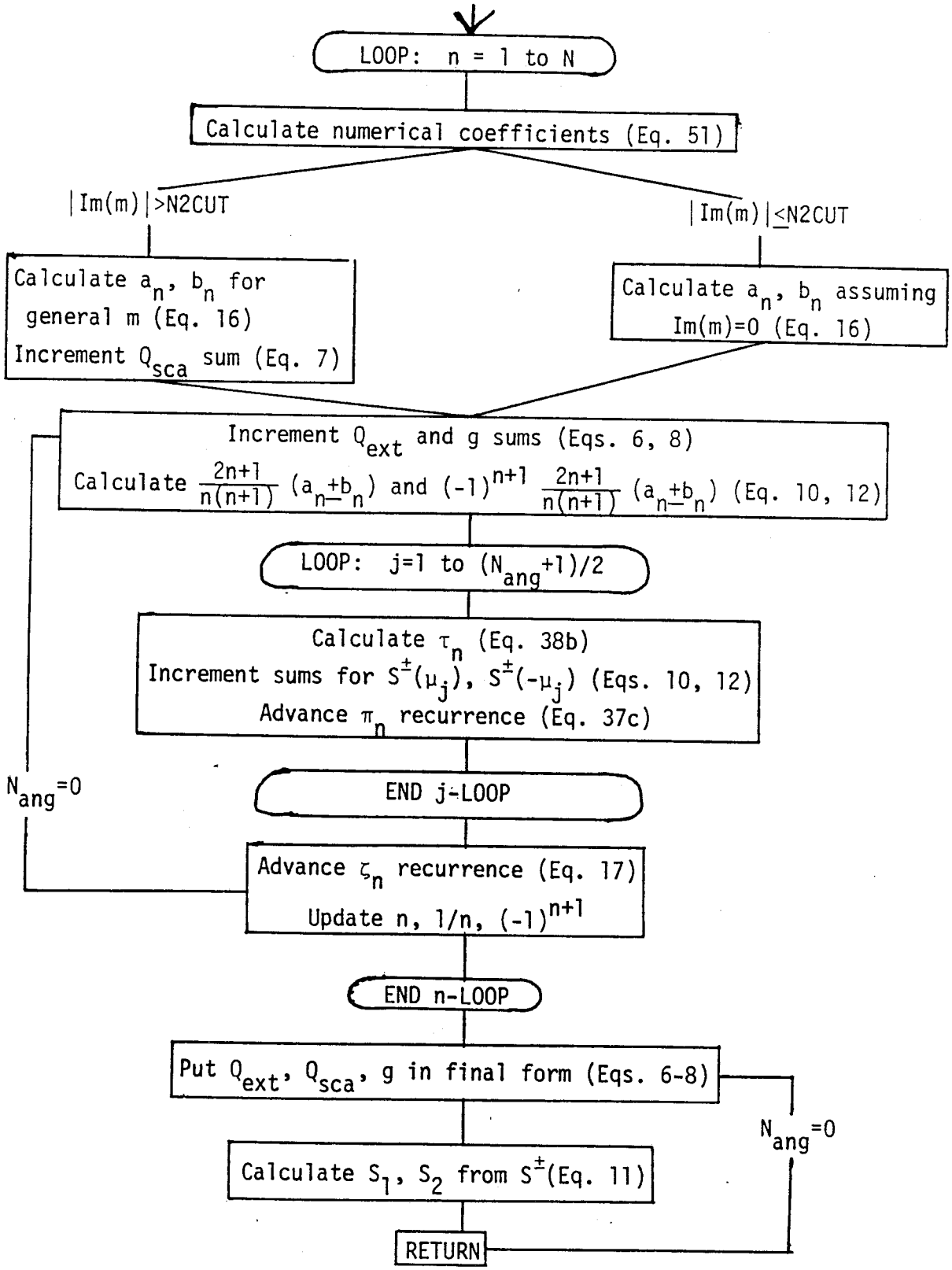
Whittaker, E. T., and G. N. Watson, 1965: A Course of Modern Analysis. Cambridge University Press, Cambridge, England, 608 pp. (4th ed., reprinted).

Wiscombe, W. J.; 1977: The Delta-Eddington Approximation for a Vertically Inhomogeneous Atmosphere. NCAR Tech. Note TN-121+STR.

Zerull, R. H., 1976: Scattering measurements of dielectric and absorbing nonspherical particles. Beitr. Phys. Atmos., 49, 166-188.

APPENDIX I. MIEVO FLOW CHART AND LISTING




LOOP: n = 1 to N
Calculate numerical coefficients (Eq. 51)
|Im(m)|>N2CUT
|Im(m)|≤N2CUT
Calculate a_n, b_n for general m (Eq. 16)
Increment Q_{sca} sum (Eq. 7)
Calculate a_n, b_n assuming Im(m)=0 (Eq. 16)
Increment Q_{ext} and g sums (Eqs. 6, 8)
Calculate $\frac{2n+1}{n(n+1)}(a_n+b_n)$ and $(-1)^{n+1}\frac{2n+1}{n(n+1)}(a_n+b_n)$ (Eq. 10, 12)
LOOP: j=1 to $(N_{ang}+1)/2$
Calculate τ_n (Eq. 38b)
Increment sums for $S^{\pm}(\mu_j)$, $S^{\pm}(-\mu_j)$ (Eqs. 10, 12)
Advance π_n recurrence (Eq. 37c)
N_{ang}=0
END j-LOOP
Advance τ_n recurrence (Eq. 17)
Update n, 1/n, $(-1)^{n+1}$
END n-LOOP
Put Q_{ext}, Q_{sca}, g in final form (Eqs. 6-8)
N_{ang}=0
Calculate S_1, S_2 from $S^{\pm}$(Eq. 11)
RETURN

```
CRAY FORTRAN COMPILER VERSION 1.04  12/27/78
COMPILATION DATE AND TIME           02/09/79    -   22:14:30
 1.       SUBROUTINE MIEV0
    C
    C  COMPUTES MIE SCATTERING  AND EXTINCTION EFFICIENCIES, ASYMMETRY
    C  FACTOR, AND ANGULAR SCATTERING FUNCTIONS
    C
    C     VERSION 0-- STRUCTURED IN ORDER TO USE THE ABSOLUTE MINIMUM AMOUNT
    C                 OF COMPUTER MEMORY, WHILE STILL VECTORIZING LOOPS
    C                 OVER SCATTERING ANGLE (WHICH, WHEN UNVECTORIZED, CAN
    C                 ACCOUNT FOR THE LION*S SHARE OF COMPUTING TIME)
    C
    C     THIS CODE IS DOCUMENTED IN-- *MIE SCATTERING CALCULATIONS--
    C     IMPROVEMENTS IN TECHNIQUE AND FAST, VECTOR-SPEED COMPUTER CODES*,
    C     BY WARREN J. WISCOMBE, NCAR TECH NOTE (1979)
    C
    C
    C   I N P U T   V A R I A B L E S
    C
    C       XX          MIE SIZE PARAMETER (2*PI*RADIUS/WAVELENGTH)
    C       IOR         COMPLEX REFRACTIVE INDEX (IMAG PART MUST BE NEGATIVE)
    C       M2CUT       THE MAGNITUDE OF THE IMAGINARY REFRACTIVE INDEX, BELOW
    C                   WHICH IT IS REGARDED AS ZERO (THE COMPUTATION PROCEEDS
    C                   FASTER FOR ZERO IMAGINARY INDEX)
    C       NUMANG      NUMBER OF ANGLES AT WHICH SCATTERING FUNCTIONS
    C                   ARE TO BE EVALUATED.  NUMANG=0 SIGNALS THAT ONLY QEXT,
    C                   QSCA, AND GFAC (SEE OUTPUT QUANTITIES BELOW) ARE TO BE
    C                   EVALUATED.  ALL POSITIVE INTEGRAL VALUES OF NUMANG ARE
    C                   PERMITTED, BUT IF IT IS ODD, 90 DEGREES MUST BE ONE OF
    C                   THE ANGLES (IF NUMANG=1, 90 DEGREES MUST BE ONLY ANGLE)
    C       XMU(N),N=1 TO NUMANG  COSINES OF ANGLES AT WHICH SCATTERING
    C                   FUNCTIONS ARE TO BE EVALUATED--THE ANGLES MUST BE
    C                   MONOTONE INCREASING AND MUST BE MIRROR
    C                   SYMMETRIC ABOUT PI/2, SO THAT IF PI/2-A IS AN ANGLE
    C                   THEN SO IS PI/2+A, WHERE A LIES BETWEEN ZERO AND PI/2
    C
    C
    C  O U T P U T   V A R I A B L E S
    C
    C       QEXT        EXTINCTION EFFICIENCY FACTOR, DEFINED AS
    C         (2/XX**2)*SUM( (2N+1)*RE(A(N)+B(N)) )
    C       QSCA        SCATTERING EFFICIENCY FACTOR, DEFINED AS
    C         (2/XX**2)*SUM( (2N+1)*(CABS(A(N))**2+CABS(B(N))**2) )
    C                   WHERE A(N) AND B(N) ARE THE USUAL MIE COEFFICIENTS
    C                   LITTLE-A-SUB-N AND LITTLE-B-SUB-N
    C       GFAC        ASYMMETRY FACTOR, DEFINED AS
    C     (4/XX**2)*SUM( (N-1/N)*RE(A(N-1)*CONJG(A(N))+B(N-1)*CONJG(B(N)))
    C                     +(2N+1)/(N(N+1))*RE(A(N)*CONJG(B(N))) ) / QSCA
    C       S1(N),S2(N),N=1 TO NUMANG   AT EACH ANGLE SPECIFIED BY XMU ARRAY,
    C                   THESE ARE THE USUAL MIE ANGULAR SCATTERING FUNCTIONS
    C                   BIG-S-SUB-ONE AND BIG-S-SUB-TWO, DEFINED AS
    C                     S1=SUM( (2N+1)/(N(N+1))*(A(N)*PI(N)+B(N)*TAU(N)) )
    C                     S2=SUM( (2N+1)/(N(N+1))*(A(N)*TAU(N)+B(N)*PI(N)) )
    C                   AND WHERE PI(N) AND TAU(N) ARE THE USUAL MIE QUANTITIES
    C                   LITTLE-PI-SUB-N AND LITTLE-TAU-SUB-N
    C
    C       --NOTE--THE PHASE FUNCTION, OR ANGULAR GAIN, FOR A PARTICULAR
    C                 MIE SIZE PARAMETER IS OBTAINED BY MULTIPLYING
    C                       I1+I2 = CABS(S1)**2+CABS(S2)**2
    C                 BY 2/XX**2.  HOWEVER, IT IS I1+I2, NOT THIS PHASE FCN,
    C                 WHICH MUST BE INTEGRATED OVER SIZES WHEN A SIZE DISTRIBU-
    C                 TION IS INVOLVED.  THIS INTEGRAL MUST THEN BE NORMALIZED
    C                 TO GIVE THE CORRECT PHASE FUNCTION.  SIMILARLY, IT IS THE
```

```
C               CROSS-SECTIONS, PROPORTIONAL TO (XX**2/2) TIMES QEXT AND
C               QSCA, WHICH SHOULD BE INTEGRATED OVER SIZES, NOT QEXT AND
C               QSCA THEMSELVES.
C
C   I N T E R N A L   V A R I A B L E S
C
C  MT1              PARAMETER. MAX. POSSIBLE NO. OF TERMS IN MIE SERIES.
C  MANGL2           PARAMETER. MAX. POSSIBLE NO. OF ANGLES FROM 0 TO
C                   90 DEGREES.
C  MM2              NUMANG+1
C  MM               MM2/2--NO. OF ANGLES FROM 0 TO 90 DEGREES
C  NOANGS           TRUE, SKIP CALCULATION OF S1,S2
C  NOIMAG           TRUE, ASSUME IMAGINARY REFRACTIVE INDEX IS ZERO AND
C                   TAKE COMPUTATIONALLY FASTER BRANCHES IN CODE
C  AIMIOR           MAGNITUDE OF IMAGINARY REFRACTIVE INDEX
C  REIOR            REAL PART OF REFRACTIVE INDEX
C  A1,A2,B1         MIE COEFFICIENTS LITTLE-A-SUB-1,LITTLE-A-SUB-2,
C                   LITTLE-B-SUB-1, WITH NUMERATOR AND DENOMINATOR
C                   EXPANDED IN POWERS OF XX, AND A FACTOR OF XX**3 DIVI-
C                   DED OUT (USED IN SMALL-XX LIMIT)
C  IORSQ,RAT,TEM    TEMPORARY VARIABLES USED IN SMALL-XX LIMIT
C  MT               NO. OF TERMS IN MIE SERIES
C  MTP1             MT+1
C  XINV             1/XX
C  ZINV             1/(IOR*XX)
C  F                ARITHMETIC STATEMENT FUNCTION USED IN DETERMINING
C                   WHETHER TO USE UP- OR DOWN-RECURRENCE FOR BIGA
C  FF               CONTAINS INTERMEDIATE AND FINAL  VALUES OF LENTZ
C                   CONTINUED FRACTION FOR BIGA(MT).  ALSO USED AS TEMPO-
C                   RARY VARIABLE IN INITIALIZING UP-RECURRENCE FOR BIGA.
C  AK,KK,DEN,DTD,   TEMPORARY VARIABLES USED IN COMPUTING FF
C  NUM,MTN,IT
C  MM               +1 AND -1 ALTERNATELY.  USED IN COMPUTING FF, SPS,SMS
C  KOUNT            ITERATION COUNTER FOR FF COMPUTATION
C  MAXIT            MAXIMUM ALLOWED NO. OF ITERATIONS IN FF COMPUTATION
C  EPS1             ILL-CONDITIONING CRITERION FOR LENTZ CONTINUED FRACN.
C  EPS2             CONVERGENCE CRITERION FOR LENTZ CONTINUED FRACTION
C  BIGA(N)          -N/Z + J-SUB-(N-1) OF Z/J-SUB-N OF Z, WHERE Z=IOR*XX
C                   AND J-SUB-N IS THE NTH-ORDER SPHERICAL BESSEL FUNCTN.
C  CTAN             EXTERNAL FUNCTION FOR COMPLEX TANGENT (USED TO INI-
C                   TIALIZE UP-RECURRENCE FOR BIGA)
C  REZ              REAL(IOR)*XX
C  REZINV           1/REZ
C  EQBIGA           REAL ARRAY EQUIVALENCED TO BIGA.  USED TO CONTAIN
C                   BIGA WHEN NOIMAG=TRUE, WHEN BIGA IS PURELY REAL.
C  IORINV           1/IOR (USED IN AN CALCULATION)
C  RIORIV           1/REAL(IOR) (USED IN AN CALCN. WHEN NOIMAG=TRUE)
C  FN               FLOATING POINT VERSION OF LOOP INDEX N IN 500-LOOP
C  FNP1             N+1
C  RN               1/N
C  RNP1             1/(N+1)
C  TWONP1           2N+1
C  COEFF            (2N+1)/(N(N+1))
C  NP1DN            (N+1)/N
C  PSIN             RICATTI-BESSEL FUNCTION PSI-SUB-(N-1) OF XX
C  PSINP1           RICATTI-BESSEL FUNCTION PSI-SUB-N OF XX
C  CHIN,CHINP1      LIKE PSIN,PSINP1 BUT FOR FUNCTION CHI
C  ZETN,ZETNP1      LIKE PSIN,PSINP1 BUT FOR FUNCTION ZETA=PSI+I*CHI
C  AN,BN            MIE COEFFICIENTS LITTLE-A-SUB-N, LITTLE-B-SUB-N
C  ANM1,BNM1        AN,BN BUT FOR (N-1) INSTEAD OF N
C  ANP              (2N+1)/(N(N+1))*(AN+BN)
C  BNP              (2N+1)/(N(N+1))*(AN-BN)
C  ANPM             (-1)**(N+1) * ANP
C  BNPM             (-1)**(N+1) * BNP
C  PIN(J),TAUN(J)   ANGULAR FUNCTIONS LITTLE-PI-SUB-N AND LITTLE-TAU-SUB-
C                   N AT JTH ANGLE
```

```
C  PINM1(J)          LITTLE-PI-SUB-(N-1) AT JTH ANGLE
C  TMP(J)            TEMPORARY ARRAY USED IN COMPUTING PIN,TAUN
C  SP(J)             S+ = S1+S2 FOR JTH ANGLE
C  SM(J)             S- = S1-S2 FOR JTH ANGLE
C  SPS(J)            S+ = S1+S2 FOR (NN2-J)TH ANGLE
C  SMS(J)            S- = S1-S2 FOR (NN2-J)TH ANGLE
C
CUSE  COMIO
C        ANGULAR DIMENSIONS
      PARAMETER (NANGL=255)
      COMPLEX IOR, S1, S2
      REAL M2CUT
      COMMON/INOUT/XX,IOR,M2CUT,NUMANG,XMU(NANGL),QEXT,QSCA,GFAC,
     *   S1(NANGL),S2(NANGL)
CEND
      PARAMETER (NANGL2=(NANGL+1)/2)
      PARAMETER (NT1=20150)
      REAL MP1DN
      LOGICAL NOIMAG, NOANGS
      COMPLEX CTAN
      COMPLEX A1,A2,B1,IORSQ,RAT
      COMPLEX ZINV,FF,AK,DEN,NUM,NTN,DTD,TT,BIGA,IORINV,AN,BN,ANM1,BNM1,
     *   ANP,BNP,ANPM,BNPM,ZET,ZETN,ZETNP1,SP,SM,SPS,SMS
      DIMENSION SP(NANGL2),SM(NANGL2),SPS(NANGL2),SMS(NANGL2),
     *   PIN(NANGL2),PINM1(NANGL2),TAUN(NANGL2),TMP(NANGL2),BIGA(NT1),
     *   EQBIGA(NT1)
      EQUIVALENCE (BIGA(1),EQBIGA(1))
      EQUIVALENCE (S1(1),PIN(1)),(S1(NANGL2),PINM1(1)),
     *            (S2(1),TAUN(1)),(S2(NANGL2),TMP(1))
      F(REM) = -15.84 + REM*(8.42+16.35*REM)
      DATA EPS1/1.E-2/, EPS2/1.E-8/
      DATA MAXIT/10000/
C
C
      IF(NUMANG.GT.NANGL .OR. NUMANG.LT.0) STOP 1000
      NN2 = NUMANG+1
      NN = NN2/2
      NOANGS = NUMANG.EQ.0
      AIMIOR = ABS(AIMAG(IOR))
      REIOR = REAL(IOR)
      NOIMAG = AIMIOR.LE.M2CUT
C
      IF(CABS(IOR)*XX.GT.0.1) GO TO 7
C
C        SMALL-PARTICLE LIMIT
C
      IORSQ = IOR**2
      RAT = (0.,0.666666666666667)*(IORSQ-1.0)
      A1 =          RAT*(1.0-0.1*XX**2+(4.*IORSQ+5.)/1400.*(XX**4))
     *     /(IORSQ+2.0+(1.0-0.7*IORSQ)*(XX**2)-(8.*IORSQ**2-385.*IORSQ
     *     +350.)/1400.*(XX**4)+(XX**3)*RAT*(1.0-0.1*XX**2))
      B1 = (XX**2/30.)*RAT*(1.0+(2.*IORSQ-5.)/70.*(XX**2))
     *     /(1.0-(2.*IORSQ-5.)/30.*(XX**2))
      A2 = (0.1*XX**2)*RAT*(1.0-XX**2/14.)
     *     /(2.*IORSQ+3.-(2.*IORSQ-7.)/14.*(XX**2))
      TEM = CABS(A1)**2+CABS(B1)**2+(5./3.)*CABS(A2)**2
      QSCA = 6.*(XX**4)*TEM
      GFAC =    REAL(A1*CONJG(A2+B1))/TEM
      QEXT = QSCA
      IF(AIMIOR.GT.1.E-12) QEXT = 6.*XX*REAL(A1+B1+(5./3.)*A2)
C
      TEM = 1.5*XX**3
      A2 = (5./3.)*A2
C   ***** VECTORIZABLE LOOP *****
      DO 5 J = 1,NN
      S1(J) = TEM*(A1+(B1+A2)*XMU(J))
```

```
      5 S2(J) = TEM*(B1+A1*XMU(J)+A2*(2.*XMU(J)**2-1.))
C     ***** VECTORIZABLE LOOP *****
      DO 6 J = 1,NN
      S1(NN2-J) = TEM*(A1-(B1+A2)*XMU(J))
    6 S2(NN2-J) = TEM*(B1-A1*XMU(J)+A2*(2.*XMU(J)**2-1.))
      RETURN
C
C
C        CALCULATE NUMBER OF TERMS IN MIE SERIES (A LEAST UPPER BOUND)
C        USING EMPIRICAL FORMULAS WHICH WERE FITTED FOR SIZE PARAMETERS
C        UP TO 20,000
C
    7 IF(XX.LE.8.0) NT = XX+4.*XX**(1./3.)+1.
      IF(XX.GT.8.0 .AND. XX.LT.4200.) NT = XX+4.05*XX**(1./3.)+2.
      IF(XX.GE.4200.) NT = XX+4.*XX**(1./3.)+2.
      NTP1 = NT+1
C        MAKE SURE ARRAY BIGA WILL BE LARGE ENOUGH
      IF(NTP1.LE.NT1) GO TO 18
      WRITE(6,8000) NT, XX
 8000 FORMAT(///* ESTIMATED LENGTH OF MIE SERIES  NT=*I6,
     *  *  FOR SIZE PARAM=*F12.2/* EXCEEDS BIGA  DIMENSIONS*)
      STOP 1001
C
C
C        COMPUTE BIGA
C
   10 XINV = 1.0/XX
      ZINV = XINV/IOR
C        DECIDE WHETHER BIGA-N CAN BE CALCULATED BY UP-RECURRENCE
      IF(AIMAG(IOR) .LT. F(REAL(IOR))/XX) GO TO 100
C
C        PREPARE FOR DOWN-RECURRENCE---
C        COMPUTE INITIAL HIGH-ORDER  BIGAN USING LENTZ METHOD
C
      FF = NTP1*ZINV
      MM = -1
      KK = 2*NT+3
      AK = (MM*KK)*ZINV
      DEN = AK
      NUM = DEN + 1.0/FF
      KOUNT = 1
C
   20 KOUNT = KOUNT+1
      IF(KOUNT.GT.MAXIT) GO TO 40
      IF(CABS(NUM/AK).GT.EPS1 .AND. CABS(DEN/AK).GT.EPS1) GO TO 30
C        ILL-CONDITIONED CASE--STRIDE TWO TERMS INSTEAD OF ONE
      MM = -MM
      KK = KK+2
      AK = (MM*KK)*ZINV
      NTN = AK*NUM + 1.0
      DTD = AK*DEN + 1.0
      FF = (NTN/DTD) * FF
      MM = -MM
      KK = KK+2
      AK = (MM*KK)*ZINV
      NUM = AK + NUM/NTN
      DEN = AK + DEN/DTD
      KOUNT = KOUNT+1
      GO TO 20
C
   30 TT = NUM/DEN
      FF = TT*FF
C        CHECK FOR CONVERGENCE
      IF(ABS(REAL(TT)-1.0).LT.EPS2 .AND. ABS(AIMAG(TT)).LT.EPS2) GOTO 50
      MM = -MM
      KK = KK+2
```

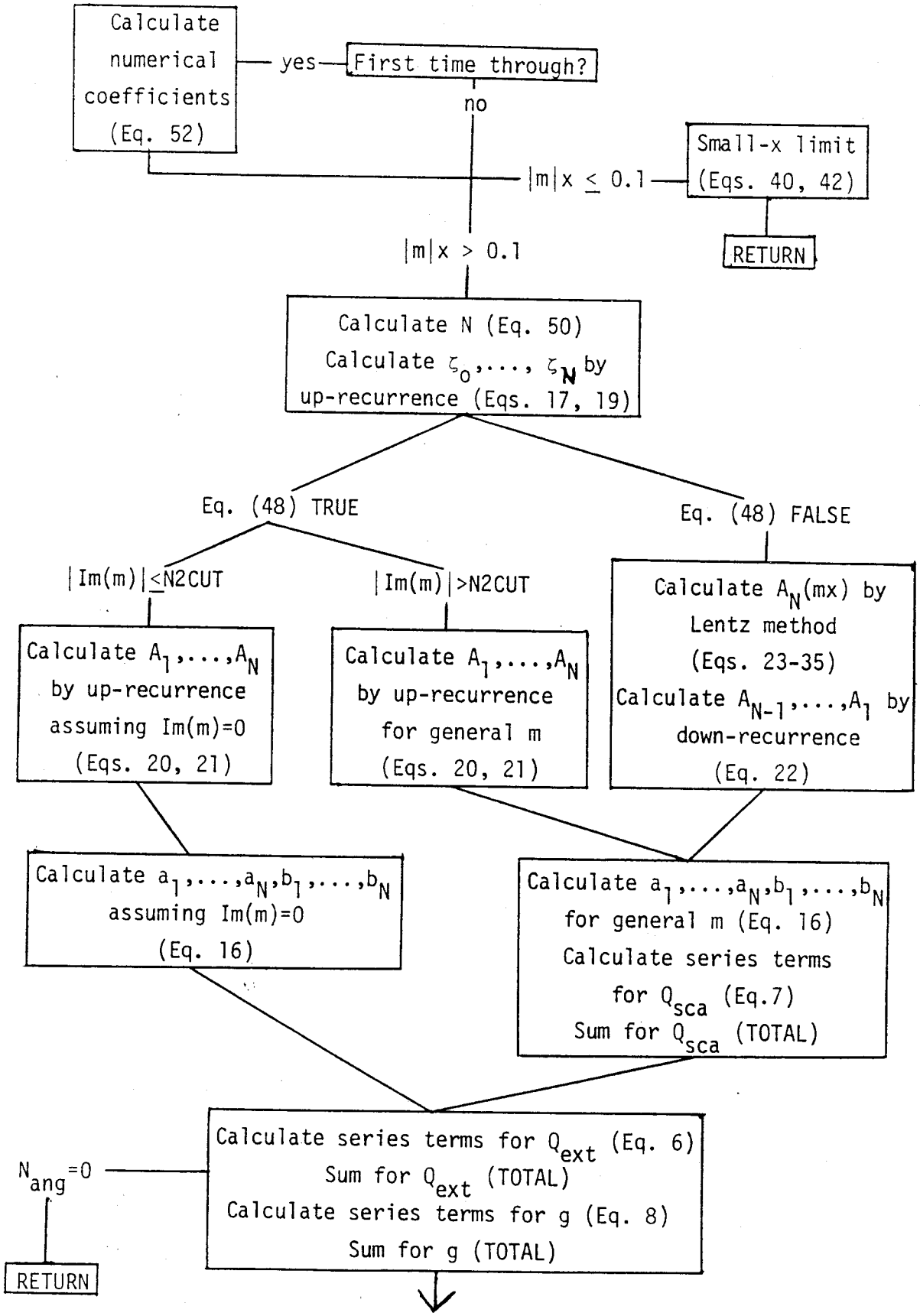
```
      AK = (MM*AK)*ZINV
      NUM = AK + 1.0/NUM
      DEN = AK + 1.0/DEN
      GO TO 20
C
   40 WRITE(6,8001) NT, XX, IOR,   AK,NUM,DEN,TT,FF
 8001 FORMAT(///* CONTINUED FRACTION FOR A-SUB-NT FAILED TO CONVERGE*/
     *    * NT=*I6/* X=*E23.8/* REFR INDEX=*2E20.8/* AK=*2E20.8/
     *    * NUM=*2E20.8/* DEN=*2E20.8/* TT=*2E20.8/* FF=*2E20.8)
      STOP 1002
C
   50 BIGA(NT) = FF
C
C        DOWNWARD RECURRENCE FOR BIGA-N
C
      DO 70 N = NT,2,-1
   70 BIGA(N-1) = (N*ZINV) - 1.0/((N*ZINV)+BIGA(N))
      GO TO 200
C
C          UPWARD RECURRENCE FOR BIGA-N
C
  100 IF(NOIMAG) GO TO 140
C
C        GENERAL CASE
C
      FF = CTAN(IOR*XX)
      BIGA(1) = -ZINV + FF/(ZINV*FF-1.0)
      DO 120 N = 2,NT
  120 BIGA(N) = -(N*ZINV)+1.0/((N*ZINV)-BIGA(N-1))
      GO TO 200
C
C        NO-ABSORPTION CASE
C
  140 REZ = REIOR*XX
      REZINV = 1.0/REZ
      TEM = TAN(REZ)
      EQBIGA(1) = -REZINV + TEM/(REZINV*TEM-1.0)
      DO 160 N = 2,NT
  160 EQBIGA(N) = - (N*REZINV) + 1.0/((N*REZINV)-EQBIGA(N-1))
C
C
  200 CONTINUE
      IORINV = 1.0/IOR
      RIORIV = 1.0/REIOR
C        INITIALIZE QUANTITIES USED FOR EFFICIENT CALCULATION OF
C        NUMERICAL COEFFICIENTS IN MIE SERIES
      FN = 1.0
      RN = 1.0
      MM = 1
C        INITIALIZE RICATTI-BESSEL FUNCTION ZETA FOR UPWARD RECURRENCE
      PSIN = SIN(XX)
      CHIN = COS(XX)
      PSINP1 = XINV*PSIN-CHIN
      CHINP1 = XINV*CHIN+PSIN
      ZETN = CMPLX(PSIN,CHIN)
      ZETNP1 = CMPLX(PSINP1,CHINP1)
C        INITIALIZE PREVIOUS COEFFICIENTS (A-SUB-N-1, B-SUB-N-1)
C        FOR USE IN ASYMMETRY FACTOR SERIES
      ANM1   = (0.0,0.0)
      BNM1   = (0.0,0.0)
C        INITIALIZE SUMS FOR EFFICIENCIES AND ASYMMETRY FACTOR
      QEXT = 0.0
      QSCA = 0.0
      GFAC = 0.0
C        INITIALIZE ANGULAR FCN PIN AND SUMS FOR S+,S- AT ALL ANGLES
C   ***** VECTORIZABLE LOOP *****
```

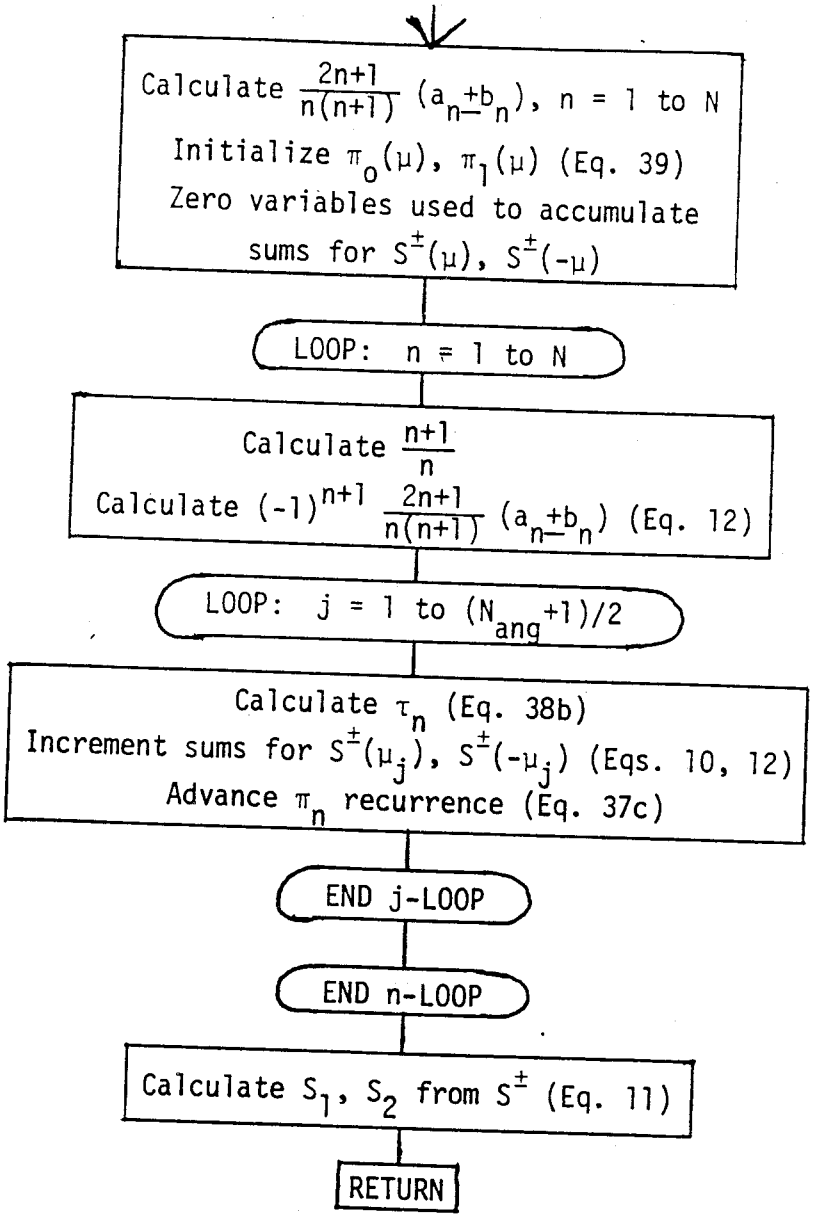
```
125.        DO 250 J = 1,MN
126.        SP(J) = (0.0,0.0)
127.        SM(J) = (0.0,0.0)
128.        SPS(J) = (0.0,0.0)
129.        SMS(J) = (0.0,0.0)
130.        PINM1(J) = 0.0
131.   250  PIN(J) = 1.0
      C
132.        DO 500 N = 1,NT
      C        COMPUTE THE VARIOUS NUMERICAL COEFFICIENTS NEEDED
133.        FNP1 = FN+1.0
134.        TWONP1 = FN+FNP1
135.        RNP1 = 1.0/FNP1
136.        COEFF = RN*RNP1
137.        NP1ON = 1.0+RN
      C
      C        CALCULATE THE MIE SERIES COEFFICIENTS LITTLE-A AND LITTLE-B
      C
138.        IF(NOIMAG) GO TO 300
      C        GENERAL CASE
139.        AN = ((IORINV*BIGA(N)+(FN*XINV))*PSINP1-PSIN)/
           *     ((IORINV*BIGA(N)+(FN*XINV))*ZETNP1-ZETN)
148.        BN = ((   IOR*BIGA(N)+(FN*XINV))*PSINP1-PSIN)/
           *     ((   IOR*BIGA(N)+(FN*XINV))*ZETNP1-ZETN)
      C        INCREMENT SERIES FOR SCATTERING EFFICIENCY
141.        QSCA   = QSCA   +  TWONP1*((REAL(AN))**2+(AIMAG(AN))**2
           *                          +(REAL(BN))**2+(AIMAG(BN))**2)
142.        GO TO 350
      C
143.   300  CONTINUE
      C        NO-ABSORPTION CASE
144.        AN = ((RIORIV*EQBIGA(N)+(FN*XINV))*PSINP1-PSIN)/
           *     ((RIORIV*EQBIGA(N)+(FN*XINV))*ZETNP1-ZETN)
145.        BN = (( REIOR*EQBIGA(N)+(FN*XINV))*PSINP1-PSIN)/
           *     (( REIOR*EQBIGA(N)+(FN*XINV))*ZETNP1-ZETN)
      C
146.   350  CONTINUE
      C        INCREMENT SERIES FOR ASYMMETRY FACTOR AND EXTINCTION EFFICIENCY
147.        GFAC = GFAC +(FN-RN)*REAL(ANM1*CONJG(AN)  BNM1*CONJG(BN))
           *           + COEFF*REAL(AN*CONJG(BN))
148.        QEXT = QEXT + TWONP1*REAL(AN+BN)
149.        IF(NOANGS) GO TO 450
      C
      C        PUT MIE SERIES COEFFICIENTS IN FORM NEEDED FOR COMPUTING S+, S-
      C
150.        ANP =      COEFF*(AN+BN)
151.        BNP =      COEFF*(AN-BN)
152.        ANPM = MM*ANP
153.        BNPM = MM*BNP
      C
      C   ***** VECTORIZABLE LOOP *****
154.        DO 400 J = 1,MM
      C        ADD UP SUMS WHILE UPWARD RECURSING ANGULAR FUNCTIONS LITTLE PI
      C        AND LITTLE TAU
155.        TMP(J) = (XMU(J)*PIN(J)) - PINM1(J)
156.        TAUN(J) =    FN*TMP(J) - PINM1(J)
157.        SP(J)  = SP(J)   +  ANP*(PIN(J)+TAUN(J))
158.        SMS(J) = SMS(J) + BNPM*(PIN(J)+TAUN(J))
159.        SM(J)  = SM(J)   +  BNP*(PIN(J)-TAUN(J))
160.        SPS(J) = SPS(J) + ANPM*(PIN(J)-TAUN(J))
161.        PINM1(J) = PIN(J)
162.        PIN(J) = (XMU(J)*PIN(J)) + NP1ON*TMP(J)
163.   400  CONTINUE
      C
      C        UPDATE RELEVANT QUANTITIES FOR NEXT PASS THROUGH LOOP
164.        MM = - MM
```

```
  165.    450 FN = FNP1
  166.        RN = RNP1
  167.        ANM1 = AN
  168.        BNM1 = BN
      C          CALCULATE RICATTI-BESSEL FUNCTIONS BY UPWARD RECURRENCE
  169.        ZET = (TWONP1*XINV)*ZETNP1-ZETN
  170.        ZETN = ZETNP1
  171.        ZETNP1 = ZET
  172.        PSIN = PSINP1
  173.        PSINP1 = REAL(ZETNP1)
  174.    500 CONTINUE
      C
      C
      C          MULTIPLY SUMS BY APPROPRIATE FACTORS TO GET QEXT, QSCA, GFAC
      C
  175.        QEXT = 2.*(XINV**2)*QEXT
  176.        QSCA = 2.*(XINV**2)*QSCA
  177.        IF(NOIMAG) QSCA = QEXT
  178.        GFAC = 4.*(XINV**2)*GFAC/QSCA
  179.        IF(NOANGS) RETURN
      C
      C
      C          RECOVER S1 AND S2 FROM S+, S-
      C
      C     ***** VECTORIZABLE LOOP *****
  180.        DO 800 J = 1,NN
  181.        S1(J) = 0.5*(SP(J)+SM(J))
  182.    800 S2(J) = 0.5*(SP(J)-SM(J))
      C
      C     ***** VECTORIZABLE LOOP *****
  183.        DO 900 J = 1,NN
  184.        S1(NN2-J) = 0.5*(SPS(J)+SMS(J))
  185.    900 S2(NN2-J) = 0.5*(SPS(J)-SMS(J))
      C
  186.        RETURN
  187.        END
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.      37, P=  121213C
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.      42, P=  121?54B
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.     109, P=  121'610
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.     146, P=  122150A
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.     180, P=  122312C
MIEV0   VECTOR BLOCK BEGINS AT SEQ. NO.     183, P=  122335C
```

```
CRAY FORTRAN COMPILER VERSION 1.05  05/04/79
COMPILATION DATE AND TIME           07/09/79    11:17:39
    1.          COMPLEX FUNCTION CTAN(Z)
    2.      C      COMPLEX TANGENT OF Z
    3.          COMPLEX Z
    4.      C
    5.          TWOX = 2.*REAL(Z)
    6.          E2Y = EXP(2.*AIMAG(Z))
    7.          E4Y = E2Y**2
    8.          DEN = 1./((2.*E2Y)*COS(TWOX)+E4Y+1.)
    9.          CTAN = CMPLX( (2.*E2Y)*SIN(TWOX)*DEN , (E4Y-1.)*DEN)
   10.          RETURN
   11.          END
```

APPENDIX II: MIEV1 FLOW CHART AND CODE LISTING




Calculate $\frac{2n+1}{n(n+1)}$ $(a_n \pm b_n)$, n = 1 to N
Initialize $\pi_0(\mu)$, $\pi_1(\mu)$ (Eq. 39)
Zero variables used to accumulate sums for $S^{\pm}(\mu)$, $S^{\pm}(-\mu)$
LOOP: n = 1 to N
Calculate $\frac{n+1}{n}$
Calculate $(-1)^{n+1}$ $\frac{2n+1}{n(n+1)}$ $(a_n \pm b_n)$ (Eq. 12)
LOOP: j = 1 to $(N_{ang}+1)/2$
Calculate τ_n (Eq. 38b)
Increment sums for $S^{\pm}(\mu_j)$, $S^{\pm}(-\mu_j)$ (Eqs. 10, 12)
Advance π_n recurrence (Eq. 37c)
END j-LOOP
END n-LOOP
Calculate S_1, S_2 from $S^{\pm}$ (Eq. 11)
RETURN

```
CRAY FORTRAN COMPILER VERSION 1.04  12/27/78
COMPILATION DATE AND TIME           02/09/79    -   22:14:31
 1.       SUBROUTINE MIEV1
      C
      C  COMPUTES MIE SCATTERING  AND EXTINCTION EFFICIENCIES, ASYMMETRY
      C  FACTOR, AND ANGULAR SCATTERING FUNCTIONS
      C
      C     VERSION 1-- STRUCTURED TO MAXIMALLY VECTORIZE LOOPS OVER THE
      C                 NUMBER OF MIE SERIES TERMS AND OVER SCATTERING ANGLE
      C                 (LARGER MEMORY REQUIREMENTS THAN VERSION V0)
      C
      C     THIS CODE IS DOCUMENTED IN-- *MIE SCATTERING CALCULATIONS--
      C     IMPROVEMENTS IN TECHNIQUE AND FAST, VECTOR-SPEED COMPUTER CODES*,
      C     BY WARREN J. WISCOMBE, NCAR TECH NOTE (1979)
      C
      C
      C   I N P U T   V A R I A B L E S
      C
      C       XX          MIE SIZE PARAMETER (2*PI*RADIUS/WAVELENGTH)
      C       IOR         COMPLEX REFRACTIVE INDEX (IMAG PART MUST BE NEGATIVE)
      C       N2CUT       THE MAGNITUDE OF THE IMAGINARY REFRACTIVE INDEX, BELOW
      C                   WHICH IT IS REGARDED AS ZERO (THE COMPUTATION PROCEEDS
      C                   FASTER FOR ZERO IMAGINARY INDEX)
      C       NUMANG      NUMBER OF ANGLES AT WHICH SCATTERING FUNCTIONS
      C                   ARE TO BE EVALUATED.  NUMANG=0 SIGNALS THAT ONLY QEXT,
      C                   QSCA, AND GFAC (SEE OUTPUT QUANTITIES BELOW) ARE TO BE
      C                   EVALUATED.  ALL POSITIVE INTEGRAL VALUES OF NUMANG ARE
      C                   PERMITTED, BUT IF IT IS ODD, 90 DEGREES MUST BE ONE OF
      C                   THE ANGLES (IF NUMANG=1, 90 DEGREES MUST BE ONLY ANGLE)
      C       XMU(N),N=1 TO NUMANG  COSINES OF ANGLES AT WHICH SCATTERING
      C                   FUNCTIONS ARE TO BE EVALUATED--THE ANGLES MUST BE
      C                   MONOTONE INCREASING AND MUST BE MIRROR
      C                   SYMMETRIC ABOUT PI/2, SO THAT IF PI/2-A IS AN ANGLE
      C                   THEN SO IS PI/2+A, WHERE A LIES BETWEEN ZERO AND PI/2
      C
      C
      C  O U T P U T   V A R I A B L E S
      C
      C       QEXT        EXTINCTION EFFICIENCY FACTOR, DEFINED AS
      C         (2/XX**2)*SUM( (2N+1)*RE(A(N)+B(N)) )
      C       QSCA        SCATTERING EFFICIENCY FACTOR, DEFINED AS
      C         (2/XX**2)*SUM( (2N+1)*(CABS(A(N))**2+CABS(B(N))**2) )
      C                   WHERE A(N) AND B(N) ARE THE USUAL MIE COEFFICIENTS
      C                   LITTLE-A-SUB-N AND LITTLE-B-SUB-N
      C       GFAC        ASYMMETRY FACTOR, DEFINED AS
      C     (4/XX**2)*SUM( (N-1/N)*RE(A(N-1)*CONJG(A(N))+B(N-1)*CONJG(B(N)))
      C                    +(2N+1)/(N(N+1))*RE(A(N)*CONJG(B(N))) ) / QSCA
      C       S1(N),S2(N),N=1 TO NUMANG   AT EACH ANGLE SPECIFIED BY XMU ARRAY,
      C                   THESE ARE THE USUAL MIE ANGULAR SCATTERING FUNCTIONS
      C                   BIG-S-SUB-ONE AND BIG-S-SUB-TWO, DEFINED AS
      C                     S1=SUM( (2N+1)/(N(N+1))*(A(N)*PI(N)+B(N)*TAU(N)) )
      C                     S2=SUM( (2N+1)/(N(N+1))*(A(N)*TAU(N)+B(N)*PI(N)) )
      C                   AND WHERE PI(N) AND TAU(N) ARE THE USUAL MIE QUANTITIES
      C                   LITTLE-PI-SUB-N AND LITTLE-TAU-SUB-N
      C
      C      --NOTE--THE PHASE FUNCTION, OR ANGULAR GAIN, FOR A PARTICULAR
      C                MIE SIZE PARAMETER IS OBTAINED BY MULTIPLYING
      C                     I1+I2 = CABS(S1)**2+CABS(S2)**2
      C                BY 2/XX**2.  HOWEVER, IT IS I1+I2, NOT THIS PHASE FCN,
      C                WHICH MUST BE INTEGRATED OVER SIZES WHEN A SIZE DISTRIBU-
      C                TION IS INVOLVED.  THIS INTEGRAL MUST THEN BE NORMALIZED
      C                TO GIVE THE CORRECT PHASE FUNCTION.  SIMILARLY, IT IS THE
      C                CROSS-SECTIONS, PROPORTIONAL TO (XX**2/2) TIMES QEXT AND
```

```
C              QSCA, WHICH SHOULD BE INTEGRATED OVER SIZES, NOT QEXT AND
C              QSCA THEMSELVES.
C
C  I N T E R N A L   V A R I A B L E S
C
C  NT1            PARAMETER. MAX. POSSIBLE NO. OF TERMS IN MIE SERIES.
C  NT12           PARAMETER = 2*NT1
C  NANGL2         PARAMETER. MAX. POSSIBLE NO. OF ANGLES FROM 0 TO
C                 90 DEGREES.
C  NN2            NUMANG+1
C  NN             NN2/2--NO. OF ANGLES FROM 0 TO 90 DEGREES
C  NOANGS         TRUE, SKIP CALCULATION OF S1,S2
C  NOIMAG         TRUE, ASSUME IMAGINARY REFRACTIVE INDEX IS ZERO AND
C                 TAKE COMPUTATIONALLY FASTER BRANCHES IN CODE
C  AIMIOR         MAGNITUDE OF IMAGINARY REFRACTIVE INDEX
C  REIOR          REAL PART OF REFRACTIVE INDEX
C  A1,A2,B1       MIE COEFFICIENTS LITTLE-A-SUB-1,LITTLE-A-SUB-2,
C                 LITTLE-B-SUB-1, WITH NUMERATOR AND DENOMINATOR
C                 EXPANDED IN POWERS OF XX, AND A FACTOR OF XX**3 DIVI-
C                 DED OUT (USED IN SMALL-XX LIMIT)
C  IORSQ,RAT,TEM  TEMPORARY VARIABLES USED IN SMALL-XX LIMIT
C  NT             NO. OF TERMS IN MIE SERIES
C  NTM1           NT-1
C  NTP1           NT+1
C  XINV           1/XX
C  ZINV           1/(IOR*XX)
C  F              ARITHMETIC STATEMENT FUNCTION USED IN DETERMINING
C                 WHETHER TO USE UP- OR DOWN-RECURRENCE FOR BIGA
C  FF             CONTAINS INTERMEDIATE AND FINAL  VALUES OF LENTZ
C                 CONTINUED FRACTION FOR BIGA(NT).  ALSO USED AS TEMPO-
C                 RARY VARIABLE IN INITIALIZING UP-RECURRENCE FOR BIGA.
C  AK,KK,DEN,DTD, TEMPORARY VARIABLES USED IN COMPUTING FF
C  NUM,NTN,TT
C  MM             +1 AND -1 ALTERNATELY.  USED IN COMPUTING FF, SFS,SMS
C  KOUNT          ITERATION COUNTER FOR FF COMPUTATION
C  MAXIT          MAXIMUM ALLOWED NO. OF ITERATIONS IN FF COMPUTATION
C  EPS1           ILL-CONDITIONING CRITERION FOR LENTZ CONTINUED FRACN.
C  EPS2           CONVERGENCE CRITERION FOR LENTZ CONTINUED FRACTION
C  BIGA(N)        -N/Z + J-SUB-(N-1) OF Z/J-SUB-N OF Z, WHERE Z=IOR*XX
C                 AND J-SUB-N IS THE NTH-ORDER SPHERICAL BESSEL FUNCTN.
C  CTAN           EXTERNAL FUNCTION FOR COMPLEX TANGENT (USED TO INI-
C                 TIALIZE UP-RECURRENCE FOR BIGA)
C  REZ            REAL(IOR)*XX
C  REZINV         1/REZ
C  EQBIGA         REAL ARRAY EQUIVALENCED TO BIGA.  USED TO CONTAIN
C                 BIGA WHEN NOIMAG=TRUE, WHEN BIGA IS PURELY REAL.
C  IORINV         1/IOR (USED IN AN CALCULATION)
C  RIORIV         1/REAL(IOR) (USED IN AN CALCN. WHEN NOIMAG=TRUE)
C  FN(N)          N (FLOATED)
C  TWONP1(N)      2N+1 (FLOATED)
C  COEFF(N)       (2N+1)/(N(N+1))
C  GCOEF(N)       N(N+2)/(N+1)
C  NP1DN          (N+1)/N
C  PSI(N)         RICATTI-BESSEL FUNCTION PSI-SUB-(N-1) OF XX
C  AN(N)          MIE COEFFICIENT LITTLE-A-SUB-N, LATER  USED TO STORE
C                 (2N+1)/(N(N+1))*(AN+BN)
C  BN(N)          FIRST USED FOR RICCATI-BESSEL FUNCTION ZETA-SUB-(N-1)
C                 OF XX, THEN FOR MIE COEFFICIENT LITTLE-B-SUB-N,
C                 FINALLY FOR (2N+1)/(N(N+1))*(AN-BN)
C  ANPM           (-1)**(N+1) * AN(N), WHERE AN(N) IS IN FINAL FORM
C  BNPM           (-1)**(N+1) * BN(N), WHERE BN(N) IS IN FINAL FORM
C  PIN(J),TAUN(J) ANGULAR FUNCTIONS LITTLE-PI-SUB-N AND LITTLE-TAU-SUB-
C                 N AT JTH ANGLE
C  PINM1(J)       LITTLE-PI-SUB-(N-1) AT JTH ANGLE
C  TMP(J)         TEMPORARY ARRAY USED IN COMPUTING PIN,TAUN
C  SP(J)          S+ = S1+S2 FOR JTH ANGLE
```

```
C  SM(J)              S- = S1-S2 FOR JTH ANGLE
C  SPS(J)             S+ = S1+S2 FOR (NN2-J)TH ANGLE
C  SMS(J)             S- = S1-S2 FOR (NN2-J)TH ANGLE
C
CUSE  COMIO
C        ANGULAR DIMENSIONS
      PARAMETER (NANGL=255)
      COMPLEX IOR, S1, S2
      REAL N2CUT
      COMMON/INOUT/XX,IOR,N2CUT,NUMANG,XMU(NANGL),QEXT,QSCA,GFAC,
     *   S1(NANGL),S2(NANGL)
CEND
      PARAMETER (NANGL2=(NANGL+1)/2)
      PARAMETER (NT1=5100)
      PARAMETER (NT12=2*NT1)
      COMMON/WJWV/FN(NT1),TWONP1(NT1),COEFF(NT1),GCOEF(NT1)
      REAL NP1ON
      LOGICAL NOIMAG
      COMPLEX CTAN
      COMPLEX A1,A2,B1,IORSQ,RAT
      COMPLEX SP,SM,SPS,SMS,ANPM,BNPM,AN,BN,BIGA,ZETA,
     * IORINV,AK,NUM,DEN,FF,ZINV,NTN,DTD,TT
      DIMENSION SP(NANGL2),SM(NANGL2),SPS(NANGL2),SMS(NANGL2),
     *   PIN(NANGL2),PINM1(NANGL2),TAUN(NANGL2),TMP(NANGL2),AN(NT1),
     *   BN(NT1),BIGA(NT1),EQBIGA(NT12),PSI(NT1)
      EQUIVALENCE  (BIGA,EQBIGA)
      EQUIVALENCE (S1(1),PIN(1)),(S1(NANGL2),PINM1(1)),
     *            (S2(1),TAUN(1)),(S2(NANGL2),TMP(1))
      F(REM) = -15.84 + REM*(0.42+16.35*REM)
      DATA EPS1/1.E-2/, EPS2/1.E-8/
      DATA MAXIT/10800/
      DATA IPASS/0/
C
      IF(IPASS.NE.0) GO TO 2
      IPASS = 1
C
C        CALCULATE NECESSARY NUMERICAL COEFFICIENTS FOR MIE SERIES
C
      DO 1 N = 1,NT1
      FN(N) = N
      TWONP1(N) = N+(N+1)
      COEFF(N) = TWONP1(N)/(N*(N+1))
      GCOEF(N) = (N+1)-1.0/(N+1)
    1 CONTINUE
C
    2 IF(NUMANG.GT.NANGL .OR. NUMANG.LT.0) STOP 1000
      NN2 = NUMANG+1
      NN = NN2/2
      AIMIOR = ABS(AIMAG(IOR))
      REIOR = REAL(IOR)
      NOIMAG = AIMIOR.LE.N2CUT
C
      IF(CABS(IOR)*XX.GT.0.1) GO TO 7
C
C         SMALL-PARTICLE LIMIT
C
      IORSQ = IOR**2
      RAT = (0.,0.666666666666667)*(IORSQ-1.0)
      A1 =          RAT*(1.0-0.1*XX**2+(4.*IORSQ+5.)/1400.*(XX**4))
     *    /(IORSQ+2.0+(1.0-0.7*IORSQ)*(XX**2)-(8.*IORSQ**2-385.*IORSQ
     *    +350.)/1400.*(XX**4)+(XX**3)*RAT*(1.0-0.1*XX**2))
      B1 = (XX**2/30.)*RAT*(1.0+(2.*IORSQ-5.)/70.*(XX**2))
     *    /(1.0-(2.*IORSQ-5.)/30.*(XX**2))
      A2 = (0.1*XX**2)*RAT*(1.0-XX**2/14.)
     *    /(2.*IORSQ+3.-(2.*IORSQ-7.)/14.*(XX**2))
      TEM = CABS(A1)**2+CABS(B1)**2+(5./3.)*CABS(A2)**2
```

```
      QSCA = 6.*(XX**4)*TEM
      GFAC =    REAL(A1*CONJG(A2+B1))/TEM
      QEXT = QSCA
      IF(AIMIOR.GT.1.E-12) QEXT = 6.*XX*REAL(A1+B1+(5./3.)*A2)
C
      TEM = 1.5*XX**3
      A2 = (5./3.)*A2
C     ***** VECTORIZABLE LOOP *****
      DO 5 J = 1,NN
      S1(J) = TEM*(A1+(B1+A2)*XMU(J))
    5 S2(J) = TEM*(B1+A1*XMU(J)+A2*(2.*XMU(J)**2-1.))
C     ***** VECTORIZABLE LOOP *****
      DO 6 J = 1,NN
      S1(NN2-J) = TEM*(A1-(B1+A2)*XMU(J))
    6 S2(NN2-J) = TEM*(B1-A1*XMU(J)+A2*(2.*XMU(J)**2-1.))
      RETURN
C
C
C        CALCULATE NUMBER OF TERMS IN MIE SERIES (A LEAST UPPER BOUND)
C        USING EMPIRICAL FORMULAS WHICH WERE FITTED FOR SIZE PARAMETERS
C        UP TO 20,000
C
    7 IF(XX.LE.8.0) NT = XX+4.*XX**(1./3.)+1.
      IF(XX.GT.8.0 .AND. XX.LT.4200.) NT = XX+4.05*XX**(1./3.)+2.
      IF(XX.GE.4200.) NT = XX+4.*XX**(1./3.)+2.
      NTP1 = NT+1
      NTM1 = NT-1
C        MAKE SURE ARRAYS AN,BN,ETC. WILL BE LARGE ENOUGH
      IF(NTP1.LE.NT1) GO TO 10
      WRITE(6,8000) NT, XX
 8000 FORMAT(///* ESTIMATED LENGTH OF MIE SERIES  NT=*I6,
     *  *   FOR SIZE PARAM=*F12.2/* EXCEEDS ARRAY DIMENSIONS*)
      STOP 1001
C
C        CALCULATE RICATTI-BESSEL FUNCTIONS BY UPWARD RECURRENCE
C
   10 XINV = 1./XX
      SX = SIN(XX)
      CX = COS(XX)
      BN(1)   =  CMPLX(SX,CX)
      BN(2)   =  CMPLX(XINV*SX-CX,XINV*CX+SX)
      DO 15 N = 1,NTM1
   15 BN(N+2) = (TWONP1(N)*XINV)*BN(N+1) - BN(N)
C ****** VECTORIZABLE LOOP  *********
      DO 16 N = 1,NTP1
   16 PSI(N) = REAL(BN(N))
C
C
C        COMPUTE BIGA
C
      ZINV = XINV/IOR
C        DECIDE WHETHER BIGA-N CAN BE CALCULATED BY UP-RECURRENCE
      IF(AIMIOR .LT. F(REIOR)/XX) GO TO 100
C
C        PREPARE FOR DOWN-RECURRENCE---
C        COMPUTE INITIAL HIGH-ORDER  BIGAN USING LENTZ METHOD
C
      FF = NTP1*ZINV
      MM = -1
      KK = 2*NT+3
      AK = (MM*KK)*ZINV
      DEN = AK
      NUM = DEN + 1.0/FF
      KOUNT = 1
C
   20 KOUNT = KOUNT+1
```

```
      IF(KOUNT.GT.MAXIT) GO TO 40
      IF(CABS(NUM/AK).GT.EPS1 .AND. CABS(DEN/AK).GT.EPS1) GO TO 30
C          ILL-CONDITIONED CASE--STRIDE TWO TERMS INSTEAD OF ONE
      MM = -MM
      KK = KK+2
      AK = (MM*KK)*ZINV
      NTN = AK*NUM + 1.0
      DTD = AK*DEN + 1.0
      FF = (NTN/DTD) * FF
      MM = -MM
      KK = KK+2
      AK = (MM*KK)*ZINV
      NUM = AK + NUM/NTN
      DEN = AK + DEN/DTD
      KOUNT = KOUNT+1
      GO TO 20
C
   30 TT = NUM/DEN
      FF = TT*FF
C        CHECK FOR CONVERGENCE
      IF(ABS(REAL(TT)-1.0).LT.EPS2 .AND. ABS(AIMAG(TT)).LT.EPS2) GOTO 50
      MM = -MM
      KK = KK+2
      AK = (MM*KK)*ZINV
      NUM = AK + 1.0/NUM
      DEN = AK + 1.0/DEN
      GO TO 20
C
   40 WRITE(6,8001) NT, XX, IOR,    AK,NUM,DEN,TT,FF
 8001 FORMAT(///* CONTINUED FRACTION FOR A-SUB-NT FAILED TO CONVERGE*/
     *    * NT=*I6/* X=*E20.8/* REFR INDEX=*2E20.8/* AK=*2E20.8/
     *    * NUM=*2E20.8/* DEN=*2E20.8/* TT=*2E20.8/* FF=*2E20.8)
      STOP 1002
C
   50 BIGA(NT) = FF
C
C        DOWNWARD RECURRENCE FOR BIGA-N
C
      DO 70 N = NT,2,-1
   70 BIGA(N-1) = (FN(N)*ZINV) - 1.0/((FN(N)*ZINV)+BIGA(N))
      GO TO 180
C
C          UPWARD RECURRENCE FOR BIGA-N
C
  100 IF(NOIMAG) GO TO 140
C
C        GENERAL CASE
C
      FF = CTAN(IOR*XX)
      BIGA(1) = -ZINV + FF/(ZINV*FF-1.0)
      DO 120 N = 2,NT
  120 BIGA(N) = -(N*ZINV)+1.0/((N*ZINV)-BIGA(N-1))
      GO TO 180
C
C        NO-ABSORPTION CASE
C
  140 REZ = REIOR*XX
      REZINV = 1.0/REZ
      TEM = TAN(REZ)
      EQBIGA(1) = -REZINV + TEM/(REZINV*TEM-1.0)
      DO 160 N = 2,NT
  160 EQBIGA(N) = - (FN(N)*REZINV) + 1.0/((FN(N)*REZINV)-EQBIGA(N-1))
C
C        CALCULATE THE MIE SERIES COEFFICIENTS LITTLE-A AND LITTLE-B
C        FOR THE NO-ABSORPTION CASE
C
```

```
127.        RIORIV = 1.0/REIOR
      C   ***** VECTORIZABLE LOOP *****
128.        DO 170 N = 1,NT
129.        AN(N)=((RIORIV*EQBIGA(N)+(XINV*FN(N)))*PSI(N+1) - PSI(N))
           *     /((RIORIV*EQBIGA(N)+(XINV*FN(N)))* BN(N+1) -  BN(N))
130.        BN(N)=(( REIOR*EQBIGA(N)+(XINV*FN(N)))*PSI(N+1) - PSI(N))
           *     /(( REIOR*EQBIGA(N)+(XINV*FN(N)))* BN(N+1) -  BN(N))
131.    170 CONTINUE
132.        GO TO 200
      C
      C         CALCULATE THE MIE SERIES COEFFICIENTS LITTLE-A AND LITTLE-B
      C         FOR THE GENERAL CASE
      C
133.    180 CONTINUE
134.        IORINV = 1.0/IOR
      C   ***** VECTORIZABLE LOOP *****
135.        DO 190 N = 1,NT
136.        AN(N)=((IORINV*BIGA(N)+(XINV*FN(N)))*PSI(N+1)-PSI(N))
           *     /((IORINV*BIGA(N)+(XINV*FN(N)))* BN(N+1)- BN(N))
137.        BN(N)=((   IOR*BIGA(N)+(XINV*FN(N)))*PSI(N+1)-PSI(N))
           *     /((   IOR*BIGA(N)+(XINV*FN(N)))* BN(N+1)- BN(N))
138.    190 CONTINUE
      C
      C         CALCULATE TERMS OF SERIES FOR SCATTERING EFFICIENCY AND SUM
      C
      C  ***** VECTORIZABLE LOOP *****
139.        DO 195 N = 1,NT
140.    195 EQBIGA(N) = TWONP1(N)*(REAL(AN(N))**2+AIMAG(AN(N))**2
           *                      +REAL(BN(N))**2+AIMAG(BN(N))**2)
      C
141.        QSCA = 2.*(XINV**2)*TOTAL(NT,EQBIGA)
      C
      C
      C         CALCULATE TERMS OF SERIES FOR EXTINCTION EFFICIENCY AND SUM
      C
142.    200 CONTINUE
      C  ***** VECTORIZABLE LOOP *****
143.        DO 300 N = 1,NT
144.    300 EQBIGA(N) = TWONP1(N)*REAL(AN(N)+BN(N))
      C
145.        QEXT = 2.*(XINV**2)*TOTAL(NT,EQBIGA)
146.        IF(NOIMAG) QSCA = QEXT
      C
      C         CALCULATE TERMS OF SERIES FOR ASYMMETRY FACTOR AND SUM THEM
      C            (SET FOLLOWING COEFFICIENTS TO ZERO BECAUSE THEY OCCUR IN
      C             LAST TERM OF SERIES FOR ASYMMETRY FACTOR)
147.        AN(NTP1) = (0.0,0.0)
148.        BN(NTP1) = (0.0,0.0)
      C   ***** VECTORIZABLE LOOP *****
149.        DO 400 N = 1,NT
150.    400 EQBIGA(N) = COEFF(N)*REAL(AN(N)*CONJG(BN(N)))
           *          + GCOEF(N)*REAL(AN(N)*CONJG(AN(N+1))+BN(N)*CONJG(BN(N+1)))
      C
151.        GFAC = 4.*(XINV**2)*TOTAL(NT,EQBIGA)/QSCA
152.        IF(NUMANG.EQ.0) RETURN
      C
      C
      C        COMPUTE S+,S- OVER ANGULAR MESH
      C
      C
      C          PUT MIE COEFFICIENTS IN FORM NEEDED
      C  ***** VECTORIZABLE LOOP *****
153.        DO 450 N = 1,NT
154.        BIGA(N) = AN(N)
155.        AN(N) = COEFF(N)*(AN(N)+BN(N))
156.    450 BN(N) = COEFF(N)*(BIGA(N)-BN(N))
```

```
      C
      C       INITIALIZE SUMS (S+,S-) AND ANGULAR FUNCTIONS (LITTLE PI)
      C   ***** VECTORIZABLE LOOP *****
  157.        DO 500 J = 1,NN
  158.        SP(J) = (0.0,0.0)
  159.        SM(J) = (0.0,0.0)
  160.        SPS(J) = (0.0,0.0)
  161.        SMS(J) = (0.0,0.0)
  162.        PINM1(J) = 0.0
  163.  500 PIN(J) = 1.0
      C
      C       ADD UP SUMS WHILE UPWARD RECURSING ANGULAR FUNCTIONS LITTLE PI
      C       AND LITTLE TAU
  164.        MM = 1
  165.        DO 700 N = 1,NT
  166.        NP1ON = (N+1)/FLOAT(N)
  167.        ANPM = MM*AN(N)
  168.        BNPM = MM*BN(N)
  169.        MM = - MM
      C   ***** VECTORIZABLE LOOP *****
  170.        DO 600 J = 1,NN
  171.        TMP(J) = (XMU(J)*PIN(J)) - PINM1(J)
  172.        TAUN(J) = FN(N)*TMP(J) - PINM1(J)
  173.        SP(J)  = SP(J) + AN(N)*(PIN(J)+TAUN(J))
  174.        SMS(J) = SMS(J) + BNPM*(PIN(J)+TAUN(J))
  175.        SM(J)  = SM(J) + BN(N)*(PIN(J)-TAUN(J))
  176.        SPS(J) = SPS(J) + ANPM*(PIN(J)-TAUN(J))
  177.        PINM1(J) = PIN(J)
  178.        PIN(J) = (XMU(J)*PIN(J)) + NP1ON*TMP(J)
  179.  600 CONTINUE
  180.  700 CONTINUE
      C
      C
      C         RECOVER S1 AND S2 FROM S+, S-
      C
      C   ***** VECTORIZABLE LOOP *****
  181.        DO 800 J = 1,NN
  182.        S1(J) = 0.5*(SP(J)+SM(J))
  183.  800 S2(J) = 0.5*(SP(J)-SM(J))
      C
      C   ***** VECTORIZABLE LOOP *****
  184.        DO 900 J = 1,NN
  185.        S1(NN2-J) = 0.5*(SPS(J)+SMS(J))
  186.  900 S2(NN2-J) = 0.5*(SPS(J)-SMS(J))
      C
  187.        RETURN
  188.        END
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.      47, P=  110236D
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.      52, P=  110277C
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.      72, P=  110427A
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     127, P=  111047B
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     133, P=  111117B
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     139, P=  111203D
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     142, P=  111232A
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     147, P=  111256B
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     153, P=  111325D
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     157, P=  111351C
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     165, P=  111407D
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     181, P=  111475D
MIEV1   VECTOR BLOCK BEGINS AT SEQ. NO.     184, P=  111520D
```

```
 CRAY FORTRAN COMPILER VERSION 1.04  12/27/78
 COMPILATION DATE AND TIME           02/09/79    -   22:14:32
  1.       FUNCTION TOTAL(LENGTH,ARRAY)
      C
      C         PARTIALLY-VECTORIZED SUM ROUTINE
      C
      C   I N P U T
      C         ARRAY      VECTOR OF TERMS TO BE SUMMED
      C         LENGTH     NUMBER OF TERMS IN *ARRAY* TO BE SUMMED
      C   O U T P U T
      C         TOTAL = SUM(ARRAY(I)), I = 1 TO LENGTH
      C
      C         NOTE--RELIES ON THE ZERO TRIP COUNT FEATURE, WHEREBY IF A
      C               DO-LOOP IS ALREADY SATISFIED AB INITIO, IT IS SKIPPED
      C               RATHER THAN BEING EXECUTED ONCE
  2.       DIMENSION ARRAY(1), HELPER(64)
      C
  3.       IF(LENGTH.GT.64) GO TO 5
      C
  4.       TOTAL = ARRAY(1)
  5.       DO 1 N = 2,LENGTH
  6.     1 TOTAL = TOTAL + ARRAY(N)
  7.       RETURN
      C
  8.     5 CONTINUE
      C   ***** VECTORIZABLE LOOP *****
  9.       DO 10 N = 1,64
 10.    10 HELPER(N) = ARRAY(N)
      C
 11.       KOUNT = (LENGTH-64)/64
 12.       JS = 63
 13.       DO 30 J = 1,KOUNT
      C   ***** VECTORIZABLE LOOP *****
 14.       DO 20 N = 1,64
 15.    20 HELPER(N) = HELPER(N) + ARRAY(N+JS+1)
 16.    30 JS = JS+64
      C
 17.       LAST = (LENGTH-64) -64*KOUNT
      C   ***** VECTORIZABLE LOOP *****
 18.       DO 40 N = 1,LAST
 19.    40 HELPER(N) = HELPER(N) + ARRAY(LENGTH+1-N)
      C
 20.       TOTAL = HELPER(1)
 21.       DO 50 N = 2,64
 22.    50 TOTAL = TOTAL + HELPER(N)
      C
 23.       RETURN
 24.       END
OTAL   VECTOR BLOCK BEGINS AT SEQ. NO.       8, P=     143C
OTAL   VECTOR BLOCK BEGINS AT SEQ. NO.      13, P=     166A
OTAL   VECTOR BLOCK BEGINS AT SEQ. NO.      17, P=     207D
```

APPENDIX III: SAMPLE CODE RESULTS

```
1.        PROGRAM MAIN
      C      RUN TEST CASES
      CUSE  COMIO
      C      ANGULAR DIMENSIONS
2.        PARAMETER (NANGL=255)
3.        COMPLEX IOR, S1, S2
4.        REAL N2CUT
5.        COMMON/INOUT/XX,IOR,N2CUT,NUMANG,XMU(NANGL),QEXT,QSCA,GFAC,
         *    S1(NANGL),S2(NANGL)
      CEND
6.        REAL I1,I2,INTEN
      C
7.        N2CUT = 1.E-7
8.        PI = 2.*ASIN(1.0)
9.        NUMANG = 37
10.       DO 1 I = 1,NUMANG
11.     1 XMU(I) = COS((I-1)*PI/36.)
      C
12.       DO 100 NIOR = 1,2
13.       IF(NIOR.EQ.1) IOR = (1.5,0.0)
14.       IF(NIOR.EQ.2) IOR = (1.5,-0.1)
15.       DO 100 NXX = 1,4
16.       IF(NXX.EQ.1) XX = 10.
17.       IF(NXX.EQ.2) XX = 100.
18.       IF(NXX.EQ.3) XX = 1000.
19.       IF(NXX.EQ.4) XX = 5000.
20.       T1 = SECOND(DUM)
21.       CALL MIEV1
22.       T2 = SECOND(DUM)
23.       DT = T2-T1
24.       QABS = QEXT-QSCA
25.       WRITE(6,1000) XX,IOR
26.       DO 10 I = 1,NUMANG
27.       ANGLE = 180./PI*ACOS(XMU(I))
28.       I1 = (REAL(S1(I)))**2+(AIMAG(S1(I)))**2
29.       I2 = (REAL(S2(I)))**2+(AIMAG(S2(I)))**2
30.       INTEN = 0.5*(I1+I2)
31.       DEGPOL = (I2-I1)/(I2+I1)
32.    10 WRITE(6,1001) ANGLE,S1(I),S2(I),INTEN,DEGPOL
33.       WRITE(6,1002) QEXT,QSCA,QABS,GFAC,DT
34.   100 CONTINUE
      C
35.       CALL EXIT
36.  1000 FORMAT(1H1, 5X,*MIE SIZE PARAMETER =*F8.2,15X,*REFRACTIVE INDEX =*
         *    F7.3,E12.3//*  ANGLE*,11X,*S-SUB-1*,21X,*S-SUB-2*,15X,
         *    *INTENSITY*,2X,*DEG OF POLZN*)
37.  1001 FORMAT(F7.2,5E14.6,F10.4)
38.  1002 FORMAT( /29X,*EXTINCTION    SCATTERING    ABSORPTION*/
         *    7X,*EFFICIENCY FACTORS*,3F14.6/ 7X,*ASYMMETRY FACTOR =*F9.6/
         *    7X,*TIME FOR THIS CASE IN SECONDS =*E12.3)
39.       END
```

MIE SIZE PARAMETER = 10.00 REFRACTIVE INDEX = 1.500 0.000E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.720500E+02	0.416662E+01	0.720500E+02	0.416662E+01	0.520856E+04	-0.0000
5.00	0.660087E+02	0.515473E+01	0.644549E+02	0.494811E+01	0.428132E+04	-0.0239
10.00	0.502427E+02	0.704128E+01	0.455224E+02	0.633977E+01	0.234775E+04	-0.0963
15.00	0.305549E+02	0.739457E+01	0.246826E+02	0.622358E+01	0.818121E+03	-0.2080
20.00	0.131942E+02	0.431603E+01	0.985638E+01	0.303822E+01	0.149547E+03	-0.2887
25.00	0.198641E+01	-0.185793E+01	0.360781E+01	-0.272489E+01	0.139195E+02	0.4685
30.00	-0.277991E+01	-0.830916E+01	0.247116E+01	-0.841057E+01	0.768071E+02	0.0005
35.00	-0.336987E+01	-0.113351E+02	0.117432E+01	-0.107502E+02	0.128393E+03	-0.0892
40.00	-0.253896E+01	-0.886339E+01	-0.262818E+01	-0.812307E+01	0.769487E+02	-0.0767
45.00	-0.189492E+01	-0.211735E+01	-0.686188E+01	-0.189570E+01	0.293697E+02	0.7251
50.00	-0.158387E+01	0.492734E+01	-0.765188E+01	0.428609E+01	0.518546E+02	0.4834
55.00	-0.109008E+01	0.810836E+01	-0.345083E+01	0.689490E+01	0.631908E+02	-0.0592
60.00	-0.286048E+00	0.588826E+01	0.309342E+01	0.490206E+01	0.341567E+02	-0.0163
65.00	0.654093E+00	0.322004E+00	0.711954E+01	0.351235E+00	0.256714E+02	0.9793
70.00	0.899374E+00	-0.456377E+01	0.561094E+01	-0.330362E+01	0.320167E+02	0.3242
75.00	0.401284E+00	-0.572290E+01	-0.426107E-01	-0.377749E+01	0.235919E+02	-0.3951
80.00	-0.319419E+00	-0.309491E+01	-0.520331E+01	-0.144173E+01	0.194168E+02	0.5014
85.00	-0.526668E+00	0.788416E+00	-0.585633E+01	0.134234E+01	0.184988E+02	0.9514
90.00	0.785066E-01	0.306855E+01	-0.187329E+01	0.232789E+01	0.917521E+01	-0.0269
95.00	0.106543E+01	0.272211E+01	0.307975E+01	0.110941E+01	0.963035E+01	0.1127
100.00	0.159302E+01	0.923048E+00	0.482125E+01	-0.873492E+00	0.136986E+02	0.7525
105.00	0.107665E+01	-0.416668E+00	0.223259E+01	-0.184390E+01	0.485859E+01	0.7257
110.00	-0.333227E+00	-0.413821E+00	-0.194258E+01	-0.124615E+01	0.280440E+01	0.8993
115.00	-0.182747E+01	0.291170E+00	-0.369331E+01	-0.553773E-02	0.853248E+01	0.5987
120.00	-0.249662E+01	0.543603E+00	-0.138829E+01	0.583572E+00	0.439826E+01	-0.4844
125.00	-0.194660E+01	-0.505382E-02	0.275657E+01	0.178699E+00	0.570995E+01	0.3364
130.00	-0.536968E+00	-0.631817E+00	0.468748E+01	-0.338446E+00	0.113872E+02	0.9396
135.00	0.892690E+00	-0.528956E+00	0.223152E+01	0.162656E+00	0.304142E+01	0.6460
140.00	0.158304E+01	0.772064E-01	-0.277081E+01	0.182343E+01	0.675713E+01	0.6282
145.00	0.131408E+01	-0.482396E-01	-0.586876E+01	0.349851E+01	0.242055E+02	0.9286
150.00	0.488377E+00	-0.186814E+01	-0.381684E+01	0.369067E+01	0.159589E+02	0.7664
155.00	-0.166476E+00	-0.477123E+01	0.246683E+01	0.185331E+01	0.161562E+02	-0.4108
160.00	-0.539898E-01	-0.665788E+01	0.841562E+01	-0.113156E+01	0.582167E+02	0.2385
165.00	0.954050E+00	-0.558337E+01	0.948679E+01	-0.370982E+01	0.679231E+02	0.5276
170.00	0.246609E+01	-0.160106E+01	0.495289E+01	-0.491906E+01	0.286866E+02	0.6986
175.00	0.379589E+01	0.290427E+01	-0.140003E+01	-0.500206E+01	0.249122E+02	0.0830
180.00	0.432164E+01	0.486827E+01	-0.432164E+01	-0.486827E+01	0.423766E+02	0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.881999	2.881999	0.000000

ASYMMETRY FACTOR = 0.742913

MIE SIZE PARAMETER = 100.00 REFRACTIVE INDEX = 1.500 0.000E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.523597E+04	-0.367988E+03	0.523597E+04	-0.367988E+03	0.275508E+08	-0.0000
5.00	0.307726E+03	-0.113495E+03	0.290021E+03	-0.145571E+03	0.106440E+06	-0.0107
10.00	-0.239264E+03	0.294757E+02	-0.237[illegible]E+03	0.512771E+02	0.586911E+05	0.0098
15.00	0.154487E+03	-0.114137E+02	0.166159E+03	-0.825525E+01	0.258367E+05	0.0712
20.00	-0.105469E+03	0.979283E+02	-0.117159E+03	0.773710E+02	0.202131E+05	-0.0248
25.00	-0.550427E+02	-0.132865E+03	-0.457175E+02	-0.125611E+03	0.192755E+05	-0.0730
30.00	0.969291E+02	-0.860123E+02	0.831928E+02	-0.902830E+02	0.159327E+05	-0.0540
35.00	0.622037E+02	-0.208937E+02	0.724102E+02	-0.348754E+02	0.538269E+04	0.2001
40.00	0.250573E+02	-0.851053E+02	0.448460E+02	-0.863264E+02	0.866710E+04	0.0919
45.00	0.139075E+02	-0.768894E+02	-0.385530E+01	-0.684476E+02	0.540267E+04	-0.1301
50.00	-0.406514E+02	0.356781E+01	-0.601957E+02	-0.285779E+02	0.305275E+04	0.4545
55.00	-0.437978E+02	0.516446E+02	-0.305744E+02	0.685491E+02	0.510959E+04	0.1026
60.00	0.291750E+02	-0.613059E+01	0.493840E+02	0.358946E+01	0.167021E+04	0.4679
65.00	0.137713E+02	-0.476328E+02	-0.967580E+00	-0.399266E+02	0.202680E+04	-0.2130
70.00	-0.453812E+01	0.152746E+02	-0.355741E+02	0.200557E+02	0.960828E+03	0.7357
75.00	0.342630E+02	0.750377E+01	0.346817E+02	0.375986E+01	0.122361E+04	-0.0054
80.00	-0.654632E+01	0.465091E+01	-0.166931E+02	-0.211137E+02	0.394465E+03	0.8365
85.00	-0.106215E+02	0.143408E+02	-0.515120E+01	0.127718E+02	0.254063E+03	-0.2535
90.00	0.224945E+02	0.396468E+00	0.162341E+02	-0.155243E+02	0.505356E+03	-0.0016
95.00	-0.154492E+01	0.268082E+02	-0.171102E+02	-0.860385E+00	0.507283E+03	-0.4214
180.00	0.699708E+01	0.125301E+02	0.658326E+01	-0.373196E+01	0.131614E+03	-0.5649
105.00	0.164632E+02	-0.525655E+01	-0.113127E+02	-0.504378E+01	0.222694E+03	-0.3412
110.00	0.100415E+01	-0.903412E+01	-0.251551E+01	0.133751E+02	0.133922E+03	0.3830
115.00	-0.710367E+01	-0.499755E+01	0.717697E+01	-0.374958E+01	0.705029E+02	-0.0700
120.00	-0.392786E+01	0.772458E+01	0.164603E+01	0.154967E+01	0.401041E+02	-0.8726
125.00	0.885824E+01	-0.192910E+01	-0.651340E+01	-0.874028E+01	0.100503E+03	0.1822
130.00	-0.782015E+01	-0.433699E+01	0.528810E+01	0.150923E+02	0.167811E+03	0.5235
135.00	0.520687E+01	0.105927E+02	-0.321579E+01	-0.177739E+02	0.232785E+03	0.4015
140.00	-0.632058E+01	-0.111092E+02	0.343645E+01	0.176605E+02	0.243533E+03	0.3292
145.00	0.127481E+02	0.769973E+01	-0.741848E+01	-0.139270E+02	0.235398E+03	0.0578
150.00	-0.894133E+01	-0.298597E+01	0.142708E+02	0.892487E+01	0.186086E+03	0.5225
155.00	-0.267976E+01	-0.474332E+02	0.727449E+01	0.112091E+02	0.121783E+04	-0.8534
160.00	-0.419956E+02	-0.694978E+02	0.204157E+02	-0.743537E+01	0.353283E+04	-0.8664
165.00	-0.218431E+02	0.209498E+02	-0.548409E+02	0.295673E+02	0.239888E+04	0.6181
170.00	0.157429E+01	0.195988E+02	0.686334E+02	-0.188336E+02	0.272592E+04	0.8582
175.00	0.173343E+02	-0.259438E+01	0.168315E+02	-0.256006E+02	0.622951E+03	0.5069
180.00	0.407669E+02	0.517546E+02	-0.407669E+02	-0.517546E+02	0.434048E+04	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.094388	2.094388	0.000000

ASYMMETRY FACTOR = 0.818246

TIME FOR THIS CASE IN SECONDS = 0.185E-02

MIE SIZE PARAMETER = 1000.00 REFRACTIVE INDEX = 1.500 0.000E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.583486E+06	-0.836182E+04	0.583486E+06	-0.836182E+04	0.253568E+12	-0.0000
5.00	-0.160461E+04	-0.157255E+04	-0.154369E+04	-0.149165E+04	0.482785E+07	-0.0455
10.00	0.825976E+03	0.562371E+03	0.843623E+03	0.238977E+03	0.883654E+06	-0.1300
15.00	-0.115541E+04	-0.590780E+03	-0.1052C9E+04	-0.597893E+03	0.163897E+07	-0.0275
20.00	-0.650822E+02	-0.111071E+04	-0.154134E+03	-0.120901E+04	0.136169E+07	0.0909
25.00	-0.936293E+03	0.452570E+03	-0.907281E+03	0.603965E+03	0.113470E+07	0.0469
30.00	0.762978E+03	-0.619575E+03	0.710261E+03	-0.779209E+03	0.103882E+07	0.0701
35.00	-0.286335E+03	0.899976E+03	-0.136516E+03	0.974024E+03	0.929666E+06	0.0406
40.00	-0.309237E+01	-0.938773E+03	0.178768E+03	-0.902471E+03	0.863859E+06	-0.0202
45.00	-0.767837E+03	-0.479382E+03	-0.695481E+03	-0.478920E+03	0.766220E+06	-0.0694
50.00	0.471775E+02	-0.360256E+03	0.105831E+03	-0.624752E+03	0.266763E+06	0.5051
55.00	-0.517710E+03	-0.116119E+03	-0.599773E+03	0.849132E+02	0.324223E+06	0.1317
60.00	0.167631E+03	-0.505082E+03	-0.263[illegible]5E+02	-0.531009E+03	0.282935E+06	-0.0010
65.00	-0.122084E+03	-0.496715E+03	-0.205678E+03	-0.413478E+03	0.237449E+06	-0.1018
70.00	-0.232624E+03	0.338938E+03	-0.269648E+03	0.281929E+03	0.160593E+06	-0.0523
75.00	-0.369643E+02	-0.181683E+03	0.188469E+03	-0.187421E+03	0.525112E+05	0.3454
80.00	-0.730761E+02	0.274177E+03	0.135932E+03	0.242898E+03	0.789951E+05	-0.0192
85.00	0.132453E+03	-0.158605E+03	-0.483195E+02	0.127806E+03	0.306843E+05	-0.3916
90.00	0.425817E+01	-0.155554E+03	-0.157646E+03	-0.174914E+02	0.248427E+05	0.0127
95.00	-0.195876E+03	0.208590E+02	0.147814E+02	-0.623308E+02	0.214530E+05	-0.8087
100.00	-0.119744E+03	-0.735988E+02	0.360609E+02	0.141665E+02	0.106283E+05	-0.8588
105.00	-0.323489E+02	0.154797E+03	0.859902E+00	-0.739534E+02	0.152392E+05	-0.6411
110.00	-0.157275E+03	-0.899481E+02	0.786996E+02	-0.216997E+02	0.197453E+05	-0.6625
115.00	0.539420E+02	0.104371E+03	0.126442E+02	-0.654336E+02	0.912219E+04	-0.5131
120.00	-0.504571E+02	0.460275E+02	0.957405E+02	-0.215432E+02	0.714740E+04	0.3474
125.00	0.979823E+02	0.462180E+02	-0.556428E+02	-0.317100E+02	0.791914E+04	-0.4821
130.00	-0.545537E+00	0.110604E+03	-0.214498E+01	-0.824758E+02	0.952024E+04	-0.2850
135.00	0.571101E+02	-0.104578E+03	-0.512866E+02	0.718836E+02	0.109979E+05	-0.2910
140.00	0.472214E+02	-0.451077E+02	-0.798499E+02	0.519025E+02	0.666722E+04	0.3604
145.00	-0.874613E+02	0.120082E+03	0.112581E+02	-0.107353E+03	0.168603E+05	-0.3089
150.00	-0.410781E+02	0.170411E+03	-0.520950E+02	-0.943775E+02	0.211737E+05	-0.4512
155.00	-0.563705E+02	0.701334E+01	0.862957E+02	0.226274E+02	0.559288E+04	0.4230
160.00	0.435050E+03	0.182903E+03	0.928021E+02	-0.431617E+03	0.208814E+06	-0.0666
165.00	-0.648554E+02	0.505608E+03	0.518586E+03	0.802089E+02	0.267605E+06	0.0290
170.00	0.483158E+03	-0.696745E+03	0.193765E+03	0.171170E+03	0.392870E+06	-0.8299
175.00	0.314084E+03	0.363000E+03	-0.190709E+03	0.426675E+03	0.224420E+06	-0.0267
180.00	0.565233E+03	0.150209E+04	-0.565233E+03	-0.150209E+04	0.257577E+07	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.013945	2.013945	0.000000

ASYMMETRY FACTOR = 0.827882

TIME FOR THIS CASE IN SECONDS = 0.153E-01

MIE SIZE PARAMETER = 5000.00 REFRACTIVE INDEX = 1.500 0.000E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.125541E+08	-0.672294E+05	0.125541E+08	-0.672294E+05	0.157609E+15	0.0000
5.00	-0.469634E+04	0.427656E+04	-0.464324E+04	0.412631E+04	0.394653E+08	-0.0223
10.00	0.433384E+04	-0.122501E+04	0.503467E+04	0.596146E+03	0.229931E+08	0.1179
15.00	0.244022E+04	0.487061E+04	0.334939E+04	0.518764E+04	0.339037E+08	0.1247
20.00	-0.435354E+04	0.256336E+04	-0.485714E+04	0.184372E+04	0.262577E+08	0.0279
25.00	0.392553E+04	-0.234029E+04	0.397800E+04	-0.325349E+04	0.236482E+08	0.1168
30.00	0.836964E+03	0.329160E+04	0.146476E+04	0.418116E+04	0.155814E+08	0.2597
35.00	-0.196465E+04	0.555056E+04	-0.163820E+04	0.523246E+04	0.323654E+08	-0.0712
40.00	0.487725E+04	0.212349E+03	0.478911E+04	-0.277019E+03	0.234225E+08	-0.0175
45.00	0.399477E+04	-0.166766E+04	0.400900E+04	-0.917758E+03	0.178268E+08	-0.0512
50.00	0.150721E+04	-0.833769E+03	0.263199E+04	-0.189052E+04	0.673414E+07	0.5594
55.00	-0.169726E+04	0.844180E+03	-0.147159E+04	0.244346E+04	0.586469E+07	0.3873
60.00	-0.150513E+04	-0.217288E+04	-0.804638E+03	-0.255927E+04	0.709207E+07	0.0148
65.00	-0.664496E+03	0.766870E+02	-0.202667E+04	0.829233E+03	0.262123E+07	0.8293
70.00	0.309361E+03	-0.170167E+04	0.131150E+04	-0.122475E+04	0.310573E+07	0.0368
75.00	-0.121856E+04	0.928778E+02	-0.486298E+03	0.120033E+04	0.158540E+07	0.0580
80.00	0.109954E+04	-0.118332E+04	0.133103E+04	0.323004E+02	0.219096E+07	-0.1909
85.00	0.116411E+04	-0.210688E+03	0.730589E+03	0.314751E+03	0.101618E+07	-0.3772
90.00	0.762792E+03	0.466846E+03	0.384561E+03	-0.433204E+03	0.567675E+06	-0.4089
95.00	0.389870E+03	0.645984E+03	-0.253019E+03	-0.141539E+03	0.323205E+06	-0.7399
100.00	0.597623E+03	-0.265265E+03	-0.232019E+03	0.347359E+02	0.241279E+06	-0.7719
105.00	-0.498980E+03	0.399672E+03	0.262569E+03	-0.142848E+03	0.249034E+06	-0.6412
110.00	-0.309067E+02	-0.100399E+03	0.501029E+03	-0.154863E+03	0.143024E+06	0.9228
115.00	0.868559E+03	0.481020E+02	-0.232051E+03	-0.153804E+03	0.417106E+06	-0.8142
120.00	0.526004E+03	0.898112E+02	-0.422..3E+03	-0.210007E+03	0.253470E+06	-0.1234
125.00	-0.432066E+03	0.621078E+02	0.450352E+03	-0.710078E+02	0.199198E+06	0.0435
130.00	0.350757E+03	0.504302E+03	-0.203070E+03	-0.357369E+03	0.273150E+06	-0.3815
135.00	0.258610E+03	0.467868E+03	-0.246671E+03	-0.401813E+03	0.254040E+06	-0.1249
140.00	-0.553203E+02	0.460415E+03	0.205843E+03	-0.336589E+03	0.185353E+06	-0.1602
145.00	-0.445549E+03	-0.646823E+03	0.301453E+03	0.137225E+03	0.363299E+06	-0.6980
150.00	0.640398E+03	0.234848E+03	-0.279947E+03	-0.308924E+02	0.272294E+06	-0.7087
155.00	-0.463005E+03	-0.251697E+03	0.431963E+03	0.218637E+03	0.256060E+06	-0.0846
160.00	-0.238426E+04	0.425324E+04	-0.683389E+03	-0.137187E+04	0.130619E+08	-0.8202
165.00	-0.191547E+04	-0.842329E+03	0.224151E+03	-0.173647E+04	0.372204E+07	-0.1764
170.00	0.297810E+03	0.177086E+04	-0.227679E+04	0.888356E+03	0.453092E+07	0.2883
175.00	-0.574395E+03	0.535172E+03	-0.159828E+04	-0.966428E+03	0.205242E+07	0.6997
180.00	-0.145625E+05	-0.507155E+04	0.145625E+05	0.507155E+04	0.237786E+09	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.008650	2.008650	0.000000

ASYMMETRY FACTOR = 0.829592

TIME FOR THIS CASE IN SECONDS = 0.741E-01

MIE SIZE PARAMETER = 10.00 REFRACTIVE INDEX = 1.500 -0.100E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.614948E+02	-0.317799E+01	0.614948E+02	-0.317799E+01	0.379171E+04	0.0000
5.00	0.548052E+02	-0.194587E+01	0.546864E+02	-0.289914E+01	0.300320E+04	-0.0014
10.00	0.377490E+02	0.936674E+00	0.374475E+02	-0.207314E+01	0.141624E+04	-0.0068
15.00	0.176021E+02	0.354839E+01	0.174032E+02	-0.856896E+00	0.313014E+03	-0.0301
20.00	0.181377E+01	0.410068E+01	0.215939E+01	0.280384E+00	0.124235E+02	-0.6183
25.00	-0.571781E+01	0.211751E+01	-0.466013E+01	0.709651E+00	0.296988E+02	-0.2518
30.00	-0.579008E+01	-0.121935E+01	-0.442757E+01	0.132155E+00	0.273164E+02	-0.2817
35.00	-0.220285E+01	-0.382223E+01	-0.137449E+01	-0.103514E+01	0.112113E+02	-0.7359
40.00	0.115404E+01	-0.407492E+01	0.740364E+00	-0.184894E+01	0.109517E+02	-0.6378
45.00	0.236137E+01	-0.190307E+01	0.722782E+00	-0.156099E+01	0.607843E+01	-0.5132
50.00	0.166733E+01	0.121208E+01	-0.309906E+00	-0.269918E+00	0.220902E+01	-0.9235
55.00	0.320851E+00	0.328101E+01	-0.733757E+00	0.111136E+01	0.632075E+01	-0.7194
60.00	-0.693755E+00	0.314952E+01	-0.281778E-01	0.159291E+01	0.646946E+01	-0.6077
65.00	-0.112325E+01	0.117704E+01	0.100600E+01	0.921251E+00	0.225394E+01	-0.1744
70.00	-0.113609E+01	-0.117949E+01	0.125557E+01	-0.241202E+00	0.215826E+01	-0.2426
75.00	-0.859872E+00	-0.247330E+01	0.406622E+00	-0.913799E+00	0.392848E+01	-0.7454
80.00	-0.273185E+00	-0.219728E+01	-0.830355E+00	-0.629697E+00	0.299433E+01	-0.6373
85.00	0.568725E+00	-0.904841E+00	-0.144283E+01	0.220249E+00	0.163623E+01	0.3019
90.00	0.135105E+01	0.417250E+00	-0.102255E+01	0.791253E+00	0.183556E+01	-0.0893
95.00	0.162959E+01	0.111577E+01	-0.424428E-01	0.537189E+00	0.209544E+01	-0.8614
100.00	0.117126E+01	0.118164E+01	0.658108E+00	-0.351174E+00	0.166228E+01	-0.6653
105.00	0.172604E+00	0.988018E+00	0.638281E+00	-0.115970E+01	0.137914E+01	0.2706
110.00	-0.857020E+00	0.821659E+00	0.178123E+00	-0.127272E+01	0.153058E+01	0.0790
115.00	-0.144661E+01	0.658103E+00	-0.732856E-01	-0.646108E+00	0.147431E+01	-0.7132
120.00	-0.145257E+01	0.316204E+00	0.255067E+00	0.235420E+00	0.116521E+01	-0.8966
125.00	-0.107357E+01	-0.246242E+00	0.914165E+00	0.810620E+00	0.135300E+01	0.1033
130.00	-0.630302E+00	-0.824923E+00	0.128067E+01	0.877085E+00	0.174359E+01	0.3819
135.00	-0.324057E+00	-0.114622E+01	0.937182E+00	0.662859E+00	0.136626E+01	-0.0370
140.00	-0.154838E+00	-0.112897E+01	0.566968E-01	0.542568E+00	0.798073E+00	-0.6271
145.00	-0.903410E-02	-0.956978E+00	-0.740810E+00	0.695546E+00	0.974236E+00	0.0599
150.00	0.205857E+00	-0.889334E+00	-0.919354E+00	0.994698E+00	0.133396E+01	0.3753
155.00	0.497121E+00	-0.999144E+00	-0.476633E+00	0.116778E+01	0.141816E+01	0.1218
160.00	0.812356E+00	-0.110065E+01	0.709831E-01	0.104108E+01	0.148012E+01	-0.2643
165.00	0.109483E+01	-0.940082E+00	0.144218E+00	0.653092E+00	0.126487E+01	-0.6463
170.00	0.131138E+01	-0.469165E+00	-0.399742E+00	0.187150E+00	0.106733E+01	-0.8175
175.00	0.144706E+01	0.629393E-01	-0.114891E+01	-0.167099E+00	0.172292E+01	-0.2177
180.00	0.149343E+01	0.296366E+00	-0.149 .3E+01	-0.296366E+00	0.231818E+01	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.459791	1.235144	1.224646

ASYMMETRY FACTOR = 0.922350

TIME FOR THIS CASE IN SECONDS = 0.400E-03

MIE SIZE PARAMETER = 100.00 REFRACTIVE INDEX = 1.500 -0.100E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.522455E+04	-0.261694E+03	0.522455E+04	-0.261694E+03	0.273645E+08	-0.0000
5.00	0.305086E+03	0.181883E+02	0.298131E+03	0.491190E+01	0.911574E+05	-0.0247
10.00	-0.116407E+03	0.238768E+02	-0.107378E+03	0.195841E+02	0.130172E+05	-0.0848
15.00	0.447729E+02	-0.434139E+02	0.392926E+02	-0.292025E+02	0.314304E+04	-0.2375
20.00	0.228785E+01	0.423964E+02	-0.130945E+01	0.251409E+02	0.121824E+04	-0.4798
25.00	-0.310537E+02	-0.211364E+02	-0.173987E+02	-0.104328E+02	0.911319E+03	-0.5484
30.00	0.336796E+02	-0.105491E+02	0.177529E+02	-0.652844E+01	0.801673E+03	-0.5537
35.00	-0.909879E+01	0.306759E+02	-0.510582E+01	0.144508E+02	0.629347E+03	-0.6268
40.00	-0.213287E+02	-0.189674E+02	-0.713673E+01	-0.812180E+01	0.465786E+03	-0.7490
45.00	0.220539E+02	-0.140293E+02	0.681356E+01	-0.350485E+01	0.370932E+03	-0.8418
50.00	0.101882E+02	0.222762E+02	0.196302E+01	0.555878E+01	0.317391E+03	-0.8905
55.00	-0.206372E+02	0.987344E+01	-0.392682E+01	0.947194E+00	0.269848E+03	-0.9395
60.00	-0.129074E+02	-0.178185E+02	-0.612760E+00	-0.207280E+01	0.230451E+03	-0.9797
65.00	0.101149E+02	-0.173895E+02	0.874630E+00	-0.208839E+00	0.202758E+03	-0.9960
70.00	0.189202E+02	-0.134859E+01	-0.669262E+00	0.749534E+00	0.180401E+03	-0.9944
75.00	0.139393E+02	0.112806E+02	-0.187769E+01	-0.736574E+00	0.162813E+03	-0.9750
80.00	0.615211E+01	0.158596E+02	-0.161371E+01	-0.253497E+01	0.149203E+03	-0.9395
85.00	0.153726E+01	0.160982E+02	-0.893979E+00	-0.382594E+01	0.138475E+03	-0.8885
90.00	0.119224E+01	0.153732E+02	-0.850854E+00	-0.466332E+01	0.130113E+03	-0.8273
95.00	0.476301E+01	0.139482E+02	-0.217232E+01	-0.502404E+01	0.123599E+03	-0.7576
100.00	0.107168E+02	0.920299E+01	-0.489459E+01	-0.368998E+01	0.118559E+03	-0.6831
105.00	0.134863E+02	-0.154765E+01	-0.661969E+01	0.110542E+01	0.114659E+03	-0.6072
110.00	0.424798E+01	-0.123699E+02	-0.206443E+01	0.593202E+01	0.111687E+03	-0.5316
115.00	-0.112976E+02	-0.565574E+01	0.699496E+01	0.321258E+01	0.109437E+03	-0.4586
120.00	-0.334094E+01	0.117717E+02	0.200077E+01	-0.785874E+01	0.107749E+03	-0.3897
125.00	0.115402E+02	-0.283158E+01	-0.818386E+01	0.219927E+01	0.106503E+03	-0.3257
130.00	-0.102852E+02	-0.529585E+01	0.788321E+01	0.388961E+01	0.105600E+03	-0.2673
135.00	0.748036E+01	0.845910E+01	-0.610605E+01	-0.671763E+01	0.104961E+03	-0.2148
140.00	-0.715616E+01	-0.842059E+01	0.611190E+01	0.704053E+01	0.104521E+03	-0.1684
145.00	0.936308E+01	0.546682E+01	-0.827119E+01	-0.474255E+01	0.104229E+03	-0.1278
150.00	-0.105069E+02	0.182926E+01	0.955976E+01	-0.172014E+01	0.104044E+03	-0.0932
155.00	0.340806E+01	-0.994987E+01	-0.315956E+01	0.934199E+01	0.103935E+03	-0.0643
160.00	0.931348E+01	0.462414E+01	-0.895099E+01	-0.441695E+01	0.103877E+03	-0.0409
165.00	-0.221822E+01	0.100651E+02	0.215464E+01	-0.984025E+01	0.103850E+03	-0.0229
170.00	-0.941598E+01	0.402882E+01	0.931863E+01	-0.399378E+01	0.103840E+03	-0.0101
175.00	-0.100624E+02	-0.168764E+01	0.100372E+02	0.168188E+01	0.103837E+03	-0.0025
180.00	-0.954952E+01	-0.355581E+01	0.954952E+01	0.355581E+01	0.103837E+03	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.089822	1.132134	0.957688

ASYMMETRY FACTOR = 0.950392

TIME FOR THIS CASE IN SECONDS = 0.200E-02

MIE SIZE PARAMETER = 1000.00 REFRACTIVE INDEX = 1.500 -0.100E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.504926E+06	-0.721499E+04	0.504926E+06	-0.721499E+04	0.255002E+12	-0.0000
5.00	-0.102968E+04	-0.212757E+03	-0.967028E+03	-0.170704E+03	0.103489E+07	-0.0682
10.00	-0.532535E+03	0.179789E+03	-0.450678E+03	0.147884E+03	0.270449E+06	-0.1681
15.00	-0.653330E+02	0.426746E+03	-0.613507E+02	0.316835E+03	0.145265E+06	-0.2830
20.00	0.376523E+03	0.162180E+02	0.247725E+03	0.225961E+02	0.101955E+06	-0.3931
25.00	-0.235384E+03	-0.253108E+03	-0.129869E+03	-0.156365E+03	0.803929E+05	-0.4861
30.00	0.229901E+03	0.221920E+03	0.109775E+03	0.123213E+03	0.646676E+05	-0.5789
35.00	-0.285229E+03	0.822162E+02	-0.130083E+03	0.245997E+02	0.528211E+05	-0.6682
40.00	-0.217675E+03	-0.170529E+03	-0.738262E+02	-0.734196E+02	0.436516E+05	-0.7517
45.00	-0.246206E+03	-0.777271E+02	-0.713790E+02	-0.350413E+02	0.364909E+05	-0.8267
50.00	-0.144982E+02	0.241139E+03	-0.148239E+02	0.560879E+02	0.308618E+05	-0.8909
55.00	-0.503654E+02	-0.220845E+03	0.216431E+01	-0.391996E+02	0.264252E+05	-0.9417
60.00	0.163394E+03	-0.136432E+03	0.225084E+02	-0.556820E+01	0.229245E+05	-0.9765
65.00	0.142056E+02	-0.199990E+03	0.983168E+01	-0.522104E+01	0.201609E+05	-0.9939
70.00	-0.672031E+02	0.176957E+03	-0.553029E+01	-0.995057E+01	0.179799E+05	-0.9928
75.00	-0.178982E+03	-0.758155E+01	0.194174E+02	-0.715082E+01	0.162601E+05	-0.9737
80.00	-0.882483E+02	0.145254E+03	0.915704E+01	-0.290160E+02	0.149061E+05	-0.9379
85.00	0.279019E+02	-0.159229E+03	-0.271564E+00	0.394049E+02	0.138426E+05	-0.8878
90.00	0.558595E+02	-0.143671E+03	-0.116307E+02	0.460706E+02	0.130097E+05	-0.8265
95.00	-0.125419E+03	0.773792E+02	0.436895E+02	-0.330736E+02	0.123601E+05	-0.7571
100.00	-0.128432E+03	-0.588009E+02	0.575057E+02	0.212738E+02	0.118559E+05	-0.6829
105.00	-0.897252E+01	0.135447E+03	0.417347E+00	-0.671401E+02	0.114671E+05	-0.6069
110.00	-0.128778E+03	0.228475E+02	0.705380E+02	-0.160555E+02	0.111696E+05	-0.5315
115.00	0.485125E+02	0.116660E+03	-0.322957E+02	-0.698729E+02	0.109441E+05	-0.4586
120.00	-0.950241E+02	0.771016E+02	0.613435E+02	-0.530387E+02	0.107752E+05	-0.3897
125.00	0.107185E+03	0.512959E+02	-0.773297E+02	-0.346577E+02	0.106505E+05	-0.3258
130.00	0.265510E+02	0.112600E+03	-0.218797E+02	-0.851931E+02	0.105601E+05	-0.2674
135.00	0.390056E+02	-0.105972E+03	-0.300337E+02	0.856658E+02	0.104961E+05	-0.2149
140.00	0.588332E+02	-0.935446E+02	-0.486957E+02	0.795029E+02	0.104520E+05	-0.1684
145.00	-0.355876E+02	0.102416E+03	0.304963E+02	-0.983323E+02	0.104227E+05	-0.1279
150.00	0.397255E+02	0.989745E+02	-0.367520E+02	-0.899089E+02	0.104042E+05	-0.0932
155.00	-0.104967E+03	0.658197E+01	0.983951E+02	-0.659809E+01	0.103933E+05	-0.0643
160.00	0.325766E+02	0.987468E+02	-0.315295E+02	-0.947021E+02	0.103874E+05	-0.0409
165.00	-0.390503E+02	0.953805E+02	0.380246E+02	-0.932794E+02	0.103846E+05	-0.0229
170.00	0.457488E+02	-0.916294E+02	-0.452265E+02	0.907355E+02	0.103836E+05	-0.0101
175.00	-0.115368E+02	-0.101373E+03	0.115246E+02	0.101115E+03	0.103834E+05	-0.0025
180.00	0.994571E+02	0.221736E+02	-0.994571E+02	-0.221736E+02	0.103834E+05	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.019703	1.106932	0.912770

ASYMMETRY FACTOR = 0.950880

TIME FOR THIS CASE IN SECONDS = 0.179E-01

MIE SIZE PARAMETER = 5000.00 REFRACTIVE INDEX = 1.500 -0.100E+00

ANGLE	S-SUB-1		S-SUB-2		INTENSITY	DEG OF POLZN
0.00	0.125423E+08	-0.666472E+05	0.125423E+08	-0.666472E+05	0.157315E+15	-0.0000
5.00	0.222714E+04	0.190241E+04	0.200270E+04	0.169246E+04	0.772728E+07	-0.1103
10.00	-0.208972E+04	0.798514E+03	-0.172290E+04	0.620806E+03	0.417917E+07	-0.1975
15.00	-0.198699E+04	0.275570E+03	-0.147596E+04	0.152643E+03	0.311291E+07	-0.2927
20.00	0.141605E+04	0.118670E+04	0.899033E+03	0.930170E+03	0.245545E+07	-0.3902
25.00	0.365848E+03	0.167373E+04	0.153103E+03	0.996380E+03	0.197572E+07	-0.4856
30.00	-0.818828E+03	-0.136541E+04	-0.365770E+03	-0.736552E+03	0.160555E+07	-0.5788
35.00	-0.696859E+03	0.130789E+04	-0.366383E+03	0.549989E+03	0.131668E+07	-0.6680
40.00	0.120790E+04	0.670896E+03	0.420519E+03	0.306897E+03	0.109011E+07	-0.7513
45.00	0.377639E+03	-0.123406E+04	0.174189E+03	-0.357729E+03	0.911909E+06	-0.8264
50.00	-0.737055E+03	0.956727E+03	-0.220330E+03	0.189279E+03	0.771474E+06	-0.8906
55.00	-0.768225E+03	-0.832129E+03	-0.871019E+02	-0.176458E+03	0.660667E+06	-0.9414
60.00	-0.106248E+04	-0.625555E+02	-0.101042E+03	-0.581135E+02	0.573177E+06	-0.9763
65.00	0.698164E+03	-0.719402E+03	0.552444E+02	0.124093E+02	0.504089E+06	-0.9936
70.00	-0.736341E+03	-0.594638E+03	0.564372E+02	-0.115428E+02	0.449555E+06	-0.9926
75.00	-0.704422E+03	-0.553289E+03	0.100150E+03	0.270281E+02	0.406550E+06	-0.9735
80.00	0.510198E+02	-0.848290E+03	0.279151E+02	0.149684E+03	0.372691E+06	-0.9378
85.00	0.789722E+03	-0.172281E+03	-0.182738E+03	0.738612E+02	0.346095E+06	-0.8878
90.00	0.543810E+03	0.546205E+03	-0.187405E+03	-0.146093E+03	0.325266E+06	-0.8264
95.00	0.469068E+03	0.575586E+03	-0.190918E+03	-0.196543E+03	0.309020E+06	-0.7570
100.00	0.632089E+03	-0.315106E+03	-0.263196E+03	0.157238E+03	0.296412E+06	-0.6829
105.00	-0.521195E+03	0.434773E+03	0.244271E+03	-0.230296E+03	0.286689E+06	-0.6069
110.00	-0.620403E+03	0.206778E+03	0.337167E+03	-0.130982E+03	0.279248E+06	-0.5315
115.00	0.622942E+03	0.105012E+03	-0.381748E+03	-0.490048E+02	0.273608E+06	-0.4586
120.00	0.582781E+03	0.186360E+03	-0.389915E+03	-0.111223E+03	0.269384E+06	-0.3897
125.00	-0.563023E+03	0.189764E+03	0.397962E+03	-0.145440E+03	0.266266E+06	-0.3258
130.00	0.309926E+03	0.488408E+03	-0.243028E+03	-0.366541E+03	0.264005E+06	-0.2674
135.00	0.396647E+03	0.401624E+03	-0.323875E+03	-0.317995E+03	0.262403E+06	-0.2149
140.00	-0.147810E+03	0.532403E+03	0.119311E+03	-0.450626E+03	0.261300E+06	-0.1684
145.00	-0.410444E+03	-0.354151E+03	0.363689E+03	0.308185E+03	0.260568E+06	-0.1279
150.00	0.512860E+03	0.146039E+03	-0.467928E+03	-0.130004E+03	0.260105E+06	-0.0932
155.00	-0.501483E+03	-0.158270E+03	0.470863E+03	0.146344E+03	0.259832E+06	-0.0643
160.00	0.429276E+03	0.293303E+03	-0.412840E+03	-0.280406E+03	0.259684E+06	-0.0409
165.00	-0.279973E+03	-0.432636E+03	0.274284E+03	0.422422E+03	0.259616E+06	-0.0229
170.00	0.101886E+03	0.501838E+03	-0.101195E+03	-0.496708E+03	0.259591E+06	-0.0101
175.00	0.307655E+02	-0.509210E+03	-0.306023E+02	0.507929E+03	0.259585E+06	-0.0025
180.00	-0.777833E+02	0.503522E+03	0.777833E+02	-0.503522E+03	0.259585E+06	-0.0000

	EXTINCTION	SCATTERING	ABSORPTION
EFFICIENCY FACTORS	2.006775	1.099193	0.907582

ASYMMETRY FACTOR = 0.950650

TIME FOR THIS CASE IN SECONDS = 0.806E-01