ANISOTROPIC QUASIPARTICLE PROPERTIES
IN ALUMINUM

A Thesis
Submitted to the Faculty
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

by
Albert B. Meador

DARTMOUTH COLLEGE
Hanover, New Hampshire

March 1975

MASTER

NOTICE
This report was prepared as an account of work sponsored by the United States Government. Neither the United States nor the United States Energy Research and Development Administration, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness or usefulness of any information, apparatus, product or process disclosed, or represents that its use would not infringe privately owned rights.

DISTRIBUTION OF THIS DOCUMENT UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Abstract

The electron - phonon mass enhancement was calculated at various points on the aluminum Fermi surface. This surface was generated on the computer using a 4 - O.P.W. model parametrized by Ashcroft's lattice potentials. The final computer calculation used Ashcroft's form factor and a phonon spectrum using a Born - von Karman force constant model which was fitted to experimental data. Anisotropy in the mass enhancement was found to be around 25%, and to depend more on the electron mixing coefficient variation than on phonon spectrum anisotropy. Differences between the second and third zones were as pronounced as the variations on each zone, and again were identified with mixing coefficient behavior. Comparison with experimentally determined cyclotron effective masses for various orbits gave good agreement.

A similar calculation for the T^3 coefficient of the lifetime for quasiparticle excitations showed order - of - magnitude variations (at $T=0$ K) over each zone of the Fermi surface. Full calculations of τ^{-1} at various temperatures showed that identifiable T^3 regimes, extending up to 10 or 20 K, exist at points on the Fermi surface that are free - electron - like. For other points, such regimes evidently lie below 5 K. Despite difficulties with the Fermi surface data grid, the τ^{-1} results were found to be reasonably accurate for temperatures as low as 5 K. For temperatures below

about 10 K, it was demonstrated that the surface - integral calculation of τ^{-1} can be carried out over a limited part of the Fermi surface without significant loss of accuracy.

Preface

This work was carried out at Dartmouth College during the years 1971 - 1975. I am deeply indebted to my adviser, Prof. W. E. Lawrence, for constant guidance and numerous suggestions. For many constructive comments on the thesis, I am indebted too to the members of my committee. I am particularly grateful for Prof. Christy's especially careful proofreading.

In addition, I would like to thank Prof. W. M. Hartmann for providing data on the aluminum phonon models. To the staff at Kiewit Computation Center, particularly Nancy Broadhead and Steve Waite, I am grateful for many favors; their indulgence in my greed for machine time was a key factor in getting this work done.

Finally, I am pleased to acknowledge the support of the U. S. Energy Research and Development Administration, through Grant # AT (11-1)-2315.

Albert B. Meador

Dartmouth College

March, 1975

Contents

Chapter I.	Introduction	1
	Equation for λ	4
Chapter II.	Approximate Calculations of λ	8
	1. Magnitude of λ in "Jellium"	8
	2. Single O.P.W. Model in Potassium	8
Chapter III.	Exact Calculation of λ in Aluminum	14
	A. Discussion of Units	14
	B. Computer Implementation	18
	1. The Fermi Surface Coordinate Data	20
	2. Area Elements	33
	3. The Electron Velocity	35
	4. The Mixing Coefficients	37
	5. Final Data-file Format	38
	6. Calculation of $V(q)$	39
	7. The Phonon Quantities ω^{σ} and $\hat{e}^{\sigma}$	41
	8. The Matrix Element and Lattice	44
	Symmetries	
	C. Results	49
	1. Values for λ	49
	2. Comparison with Experimental Data	49
	3. Discussion of Variation in λ	52
	4. Effective Masses	56
Chapter IV.	Quasiparticle Lifetime Calculation	59
	in Aluminum	
	A. Formulas for τ^{-1}	59
	B. Results	64

C. Discussion of Results	73
1. Umklapp Effects	73
2. Regime of T^3 Behavior	77
3. The Low- T τ^{-1}/T^3 Anisotropy	78
4. Errors in the τ^{-1} Calculation	79
Appendix A Lists of Programs	81
Appendix B Errors in the λ Calculation	106
1. Coarseness of the Data Grid	106
2. The Phonon Model	113
3. Choice of the Form Factor $V(q)$	114
References	118

List of Tables

Table	Title	Page
3.1	Summary of physical and calculational units and equations for various quantities.	17
3.2	Values obtained for λ on zone 2 of the Fermi surface.	50
3.3	Values obtained for λ on zone 3 of the Fermi surface.	51
3.4	Calculated values of λ , compared with band values calculated previously by Ashcroft, and with experimental values.	57
4.1	Values of τ^{-1}/T^3 for various Fermi surface points at temperatures from 0 to 100 K.	68-69
B.1	Values of λ calculated from the old form factor and the new one.	116

List of Illustrations

Figure	Title	Page
-	Aluminum Fermi Surface, Zone 2.	ii
1.1	Examples of two quantities that depend on the quasiparticle effective mass λ , and one that does not.	3
2.1	The calculation for $\lambda(k)$ for a simple 1 - O.P.W. model.	9
2.2	λ as a function of the angle θ of the initial wavevector with the reciprocal lattice vector.	12
3.1	The Brillouin zone of aluminum	15
3.2	Flowchart of the surface integral program X-4OPW.	19
3.3	The iteration scheme used to locate roots.	22
3.4	Patches defining the second zone of the Fermi surface in aluminum.	25
3.5	Patches defining the third zone of the Fermi surface in aluminum.	30
3.6	Another view of the third-zone patches.	31
3.7	Basal plane view of third-zone arm.	32
3.8	The area element at P is found by constructing parallelograms with the nearest-neighbors A, B, C and D.	34
3.9	Pseudopotential form factor $V(q)$, showing the two choices for R_0 discussed in section 6.	40

3.10	Phonon spectra calculated from two force constant models.	43
3.11	An initial wavevector p on the symmetry plane $\Gamma L K$.	45
3.12	Remapping of the third-zone arm, showing a low- q phonon process not obviously described by the third-zone data as it is represented.	47
3.13	Locations of points on zone 2 where λ has been calculated.	50
3.14	Locations of points on zone 3 where λ has been calculated.	51
3.15	Variation in λ along the second-zone 110 section.	53
3.16	Variation in λ along the second-zone 111 and 001 sections.	54
3.17	Variation in λ along the third-zone 110, "150", and 001 sections.	55
4.1	Use of the Fermi surface data grid in the programs (a) T3 and (b) TQ-0.	62
4.2	τ^{-1}/T^3 as a function of temperature on the 110 section of zone 2.	65
4.3	τ^{-1}/T^3 as a function of temperature on the 110 section of zone 2, from X to L.	66
4.4	τ^{-1}/T^3 as a function of temperature on the 110 section of zone 3.	67

4.5	Anisotropic variation in τ^{-1}/T^3 along the [110] section of zone 2.	70
4.6	τ^{-1}/T^3 along the [110] section of zone 3.	71
4.7	τ^{-1}/T^3 along the "neck" section of zone 3.	72
4.8	Comparison of full τ^{-1}/T^3 calculation at X on zone 2 with the "reduced" calculation using only part of the Fermi surface near X.	74
4.9	Comparison of full τ^{-1}/T^3 calculation at L on zone 2 with the "reduced" calculation using only the part of the Fermi surface near L.	75
4.10	Comparison of full τ^{-1}/T^3 calculation at K on zone 2 with the "reduced" calculation using only the part of the Fermi surface near K.	76
B.1	Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector k at the point nearest K.	107
B.2	Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector k in the $k_z=0$ plane, near K.	108
B.3	Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector in the middle of this region.	109
B.4	Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector k at the point nearest L.	110

B.5	Contour plot of the λ integrand for the "X", "V1", and "V2" regions of zone 3, with the initial wavevector k at the corner of the "V2" region nearest K.	111
B.6	Contour plots of the λ integrand for the "KL" region of zone 2 with the initial wavevector k at a point near W in the $k_z=0$ plane.	112
B.7	Comparison of the two forms considered for $V(q)$.	115

I. Introduction

In the study of metals, a fundamental quantity is the band function that relates the energy of a conduction electron to its wavevector. Since electrons obey Fermi-Dirac statistics, electron states will be populated up to a non-zero energy called the Fermi energy; to this corresponds a constant-energy surface in k -space called the Fermi surface. For free electrons, $E = (\hbar^2/2m) |k|^2$ and the surface at $E = E_F$ is just a sphere. For many metals the Fermi surface closely resembles a spherical free-electron surface which has been warped and split along planes of the reciprocal lattice by Bragg scattering.

Knowledge of the detailed structure of the Fermi surface is necessary in calculating many properties of a metal. For instance, the period of a cyclotron resonance orbit depends on its size; the orbit is just a planar section of the Fermi surface with period

$$T \sim \oint_{\text{orbit}} \frac{dk}{|v_k|} \sim \frac{dA}{dE}.$$

Here the last relation makes it clear that the cross-sectional area A of the orbit must be known as a function of the band energy E in order to calculate the derivative. This relation will not in general be known unless the actual physical dimensions of the Fermi surface are known.

Another example is the density of electron states at the Fermi level; this quantity is

$$N(E_F) \sim \int \frac{dS_k}{|v_k|}, \quad \text{i.e.}$$

a surface integral of the inverse electron velocity. Neither the velocity nor the metric associated with dS can be known without first describing the structure of the Fermi surface. By contrast, a quantity like the Pauli susceptibility,

$$\chi \sim (N_+ - N_-),$$

depends on the difference in total electron populations of opposite spin; here the details of the band structure $E(k)$ at the Fermi energy cancel out in the two N 's and hence are not important. Thus χ may be calculated without reference to the Fermi surface structure. These examples are illustrated in Figure 1.1.

For the large class of properties that do depend on occupation at the Fermi energy, a quantity of considerable interest is the electron-phonon mass enhancement factor λ ; this is a measure of the scattering of an electron at a state k into all other states k' on the Fermi surface, mediated by virtual lattice phonons. It represents a perturbation-theory correction to quantities derived from first-order stationary-state models. For instance, calculation of the cyclotron resonance period can be refined via the mass enhancement factor to:

$$T \sim \oint_{\text{orbit}} \frac{dk \cdot (1+\lambda)}{|v_k|}.$$

Calculation of λ in aluminum is the major concern of this work. A series of approximations is presented, culminating in a computer calculation based on a 4-O.P.W. (orthogonalized-plane-wave) model.

(a) Cyclotron Resonance Period

$$T \sim \oint_{\text{orbit}} \frac{dk}{|v_k|}$$

(b) Electron Density of States (Specific Heat)

$$N(E_F) \sim \text{F.S.} \frac{dS_k}{|v_k|}$$

(c) Pauli Susceptibility

$$\chi \sim (N_+ - N_-)$$

Figure 1.1. Examples of two quantities that depend on the quasiparticle effective mass λ , and one that does not.

Equation for λ

A formula will be derived for λ which is based on ordinary Rayleigh-Schrödinger perturbation theory. While the resulting formula will be "correct", its interpretation in the framework of perturbation theory will require that the derivation be reconsidered as a result of the self-consistent perturbation theory of Brillouin and Wigner¹. The approach taken here, via Rayleigh - Schrödinger theory, is based on the development presented by Wilkins². For a more rigorous treatment, see Schrieffer³.

The electron - phonon interaction is described for a 1 - O.P.W. model by the Hamiltonian⁴:

$$H_{e\phi} = \sum_{kk'} \sum_{\sigma} M_{kk'}^{\sigma} (a_{-q}^{\dagger} + a_q) c_k^{\dagger} c_{k'}, \quad \text{where} \quad (1.1)$$

$$M_{kk'}^{\sigma} = i \sqrt{\frac{\hbar}{2MN\omega_q^{\sigma}}} |k-k'| V(k-k') \quad \text{and}$$

$$q = k - k'. \quad (M \text{ is the ionic mass and } N \text{ is the number of ions.})$$

To second order in perturbation theory the energy of an N - electron state will be:

$$E^N = \epsilon_0^N + \langle \phi | H_{e\phi} | \phi \rangle + \sum_n \frac{\langle \phi | H_{e\phi} | \phi_n \rangle \langle \phi_n | H_{e\phi} | \phi \rangle}{\epsilon_0^N - \epsilon_n^N}, = \epsilon_0^N + \epsilon_2^N,$$

where $\epsilon_0^N = \sum_k \epsilon_k n_k$ is the sum of unperturbed single-particle energies and where $\langle \phi | H_{e\phi} | \phi \rangle = 0$ because of the orthogonality of $|\phi\rangle$ to $a^{\dagger}|\phi\rangle$ and $a|\phi\rangle$. The second-order term ϵ_2^N has non-zero parts:

$$\epsilon_2^N = \langle \Phi | \sum_{k,k'} (|M_{kk'}|^2) [a_{-q}^\dagger c_k^\dagger c_k (\epsilon_k - H_0)^{-1} a_{-q} c_k^\dagger c_k + a_q c_k^\dagger c_k (\epsilon_k - H_0)^{-1} a_q^\dagger c_k^\dagger c_k] | \Phi \rangle \quad (1.2)$$

$$= \sum_{k,k'} |M_{kk'}|^2 \langle n_k (1-n_{k'}) \rangle \left(\frac{\langle n_{-q} \rangle}{\epsilon_k - \epsilon_{k'} + \hbar\omega_q} + \frac{\langle n_q + 1 \rangle}{\epsilon_k - \epsilon_{k'} - \hbar\omega_q} \right) \quad (1.3)$$

where the sum over polarizations σ has been suppressed for simplicity. Now restrict (1.3) to low temperatures with $\langle n_q \rangle = 0$. The total perturbed energy becomes

$$E^N = \sum_k \epsilon_k n_k + \sum_{k,k'} |M_{kk'}|^2 \langle n_k \rangle \frac{1 - \langle n_{k'} \rangle}{\epsilon_k - \epsilon_{k'} - \hbar\omega_q} \quad (1.4)$$

The energy of a quasiparticle excitation should be given by the energy difference $E_p = E^{N+1} - E^N = \epsilon_p + \epsilon_2^{N+1} - \epsilon_2^N = \epsilon_p + \Delta\epsilon_2$, i.e. the energy change due to an extra electron with momentum $\hbar p$. The second order difference gives rise to two terms -- the first being a sum over unfilled intermediate states from the new-filled state p , the second a sum over filled initial states to the previously unfilled state at p . So

$$E_p = \epsilon_p + \sum_{k'}^{\text{unfilled}} |M_{pk'}|^2 \frac{1 - \langle n_{k'} \rangle}{\epsilon_p - \epsilon_{k'} - \hbar\omega_{p-k'}} - \sum_k^{\text{filled}} |M_{kp}|^2 \frac{\langle n_k \rangle}{\epsilon_k - \epsilon_p - \hbar\omega_{k-p}} \\ = \epsilon_p + \sum_q |M_{p,p+q}|^2 \left[\frac{1 - \langle n_{p+q} \rangle}{\epsilon_p - \epsilon_{p+q} - \hbar\omega_q} + \frac{\langle n_{p+q} \rangle}{\epsilon_p - \epsilon_{p+q} + \hbar\omega_q} \right] \quad (1.5)$$

where the substitution $-q$ for q has been made in the last term. Write the q -sum as a volume integral over constant energy shells:

$$E_p = \epsilon_p + \frac{\Omega}{8\pi^3} \int \frac{dS_{k'}}{|\mathbf{v}_{k'} \cdot \mathbf{E}_{k'}|} |M_{pk'}|^2 \left[\int_0^\infty \frac{d\epsilon_{k'}}{\epsilon_p - \epsilon_{k'} + \hbar\omega_q} + \int_{-\infty}^0 \frac{d\epsilon_{k'}}{\epsilon_p - \epsilon_{k'} + \hbar\omega_q} \right] \quad (1.6)$$

or

$$E_p = \epsilon_p + \frac{\Omega}{8\pi^3} \int \frac{dS_{k'}}{|\mathbf{v}_{k'} \cdot \mathbf{E}_{k'}|} |M_{pk'}|^2 \int_0^\infty d\tilde{\epsilon}_{k'} \left(\frac{1}{\tilde{\epsilon}_p - \tilde{\epsilon}_{k'} + \hbar\omega_q} + \frac{1}{\tilde{\epsilon}_p + \tilde{\epsilon}_{k'} + \hbar\omega_q} \right) \quad (1.7)$$

where $\tilde{\epsilon} = \epsilon - E_{\text{Fermi}}$. The integration limits in (1.6) reflect the fact that the first sum was over unfilled states with $\tilde{\epsilon}_{k'} > 0$, and that the second was over filled states with $\tilde{\epsilon}_{k'} < 0$.

The effective mass can now be calculated from (1.7) by differentiating E with respect to p :

$$\begin{aligned} \hbar v_p &= \nabla_p E_p|_{\tilde{\epsilon}_p=0} = \nabla_p \tilde{\epsilon}_p \left(1 + \frac{\partial}{\partial \tilde{\epsilon}_p} \Delta \epsilon_2 \right) \\ &= v_p \tilde{\epsilon}_p (1 - \lambda), \end{aligned} \quad (1.8)$$

so that λ is defined by

$$\lambda = -\frac{\Omega}{8\pi^3} \int \frac{dS_{k'}}{|\mathbf{v}_{k'} \cdot \mathbf{E}_{k'}|} |M_{pk'}|^2 \frac{\partial}{\partial \tilde{\epsilon}_p} \int_0^\infty d\tilde{\epsilon}_{k'} \left(\frac{1}{\tilde{\epsilon}_p - \tilde{\epsilon}_{k'} + \hbar\omega_q} + \frac{1}{\tilde{\epsilon}_p + \tilde{\epsilon}_{k'} + \hbar\omega_q} \right) \Big|_{\tilde{\epsilon}_p=0}.$$

The integral over $\tilde{\epsilon}_{k'}$ can be evaluated at once, and the limit of the derivative as $\tilde{\epsilon}_p \rightarrow 0$ is just $-2/\hbar\omega_q$. Now restoring the sum over polarizations, we have the result

$$\lambda = \frac{\Omega}{4\pi^3} \int \frac{dS_{k'}}{|\mathbf{v}_{k'} \cdot \mathbf{E}_{k'}|} \sum_{\sigma} |M_{pk'}^{\sigma}|^2 \cdot \frac{1}{\hbar\omega_q^{\sigma}}. \quad (1.9)$$

It appears that λ in (1.9) is always positive, so the quasiparticle velocity v_p will always be less than the band velocity $\frac{1}{\hbar} \nabla \epsilon_p$. But with the magnitude of λ in (1.9) observed to be as large as unity, the validity of second-order perturbation theory must be questioned. Following Wilkins², we see that it is possible to insert the actual quasiparticle energy E_p in (1.5) rather than the band energy ϵ_p . This amounts to using Brillouin - Wigner perturbation theory instead of Rayleigh - Schrödinger. It does not alter the expression (1.9) for λ , but it does change the interpretation of λ ; for

then, in contrast to (1.8) we have self-consistently:

$$\begin{aligned} \hbar \nabla_p &= \nabla_p \tilde{E}_p |_{\tilde{E}_p=0} = \nabla_p \tilde{E}_p - \lambda \nabla_p \tilde{E}_p |_{\tilde{E}_p=0} , \\ \text{or } \hbar \nabla_p &= \nabla_p \tilde{E}_p \cdot \frac{1}{1+\lambda} . \end{aligned} \quad (1.10)$$

Then the mass enhancement factor is just $(1+\lambda)$.

The Orthogonalized Plane Wave Calculation

When conduction-electron states are modelled in an O.P.W. approximation, the matrix element M_{pk}^σ is a sum over terms of the form defined in (1.1) multiplied by mixing coefficients α , β of the initial and final O.P.W. states. Furthermore, a realistic phonon model in aluminum will include three modes specified by frequencies ω_q^σ and polarization vectors $\hat{e}_q^\sigma$. With all these factors inserted the complete expression for

λ becomes:

$$M_{pk}^\sigma = i \sqrt{\frac{\hbar}{2MN\omega^\sigma}} \sum_{i=1}^4 \sum_{j=1}^4 \alpha_i \beta_j \tilde{e}_{pk}^\sigma \cdot (p-k'-G_i+G_j) V(p-k'-G_i+G_j), \quad (1.11)$$

where the G 's are the 4 reciprocal lattice vectors of the 4-O.P.W. aluminum model.⁵

II. Approximate Calculation of λ

1. Magnitude of λ in "Jellium"

It is useful to estimate the magnitude of λ using a 1-O.P.W. model and ignoring the sum over polarizations.

Regrouping terms in (1.9) we obtain:

$$\lambda \equiv \left[\frac{\Omega}{4\pi^3 N} \int \frac{dS_k}{|v_k E_k|} \right] \cdot \frac{\bar{q}^2}{\omega^2} \cdot \frac{V(q)^2}{2M} = \frac{\Omega}{N} \cdot \frac{n}{\frac{2E_F}{3}} \cdot \frac{1}{v_s^2} \cdot \frac{1}{2M} \left(\frac{2E_F}{3} \right)^2,$$

making the approximations of an effective constant sound velocity v_s and replacing $V(q)$ by $\frac{2}{3}E_F$, the value for a screened ionic potential⁶ at $q=0$. In a "jellium" of uniform positive charge, $v_s^2 = (mZ/3M)v_F^2$ so that²

$$\lambda \approx \frac{1}{2}. \quad (2.1)$$

2. Single O.P.W. Model in Potassium

The alkali metals, with only one valence electron per atom, tend to have small enough "Fermi volumes" that the free-electron Fermi surface lies well inside the first Brillouin zone. Thus a 1-O.P.W. (free electron) model may be adequate to describe the Fermi surface.

Here such a model is used, not for its adequacy in reproducing the Fermi surface, but for its simplicity in calculating λ . The geometry of the model is shown in Figure 2.1. Here it is necessary to introduce a limiting Debye phonon wavevector q_D and consider final electron states k' with $|k-k'| \leq q_D$. To reach electron states outside this spherical cap it is necessary to consider the Fermi surface remapped

Figure 2.1. The calculation for $\lambda(k)$ for a simple 1-QPW model. λ is broken into parts called "normal" and "umklapp" defined by the condition $|k-k'| = q \leq q_D$.

through a reciprocal lattice vector with phonon wavevectors between the original and remapped surface. This leads to a division of the surface integral into two parts identified as "normal" and "umklapp" terms. Depending on the magnitude of q_D , the two terms will cover more or less of the total Fermi surface. The example in Fig. 2.1 shows a case (potassium) in which the normal and umklapp regions overlap, but do not succeed in covering the total Fermi surface.

The normal part of λ is computed by taking the matrix element in (1.9) equal to $i\sqrt{\frac{\hbar}{2MN\omega}} \cdot q \cdot V(0)$, with $V(0) = (2/3)E_F$, and by taking $\omega = v_s q$. Then

$$\lambda_n = \Omega \left[\frac{1}{4\pi^3} \int \frac{dS_{k'}}{|v_{k'} E_{k'}|} \right] \int \frac{d\Omega}{4\pi} \frac{\hbar}{2MN\omega} q^2 V^2(0) \frac{1}{\hbar\omega} = \frac{\Omega \mathcal{N}(E_F) V^2(0)}{2MNv_s^2} \int_0^{q_D} \frac{2\pi q dq}{4\pi k_F^2}$$

With $\mathcal{N}(E_F) = n/(\frac{2}{3} E_F)$, and taking $v_s^2 = \frac{\pi Z}{3M} v_F^2$ (the jellium sound velocity), the equation becomes

$$\lambda_n = \frac{q_D^2}{8k_F^2}$$

Note that restricting λ to this normal term and setting $q_D = 2k_F$ just gives back the value $\lambda = 1/2$.

The umklapp part of λ involves similar assumptions; again take $\omega = qv_s$, but now let $M_{kk'} = i\sqrt{\frac{\hbar}{2MN\omega}} \Pi_u$, with Π_u an average (constant) value of $qV(q)$ to be determined. Then

$$\lambda_u = \frac{\Omega \mathcal{N}(E_F)}{2MN} (\Pi_u)^2 \int \frac{dS}{4\pi k_F^2} \frac{1}{\omega^2} = \frac{\Omega n (\Pi_u)^2}{4MN (\frac{2}{3} E_F) v^2} \int_0^{\phi_{\max}} \frac{\sin \phi d\phi}{s^2 + k_F^2 - 2sk_F \cos \phi}$$

where s depends on the initial wavevector k through the formula $s^2 = G^2 + k_F^2 - 2Gk_F \cos \theta$ (see Figure 2.1). The result is

$$\lambda_u = \frac{(\Pi_u)^2}{4sk_F (E_F)^2} \cdot \ln \frac{q_D}{q_{\min}}$$

where $q_{\min}$ is a function of the initial wavevector given by $q_{\min} = s - k_F$.

Now, instead of just adding λ_n and λ_u , let us try to compensate for the irregular coverage of the Fermi surface by letting the normal term cover the whole surface (thus becoming $\lambda_n = 1/2$) and then subtracting a correction from the umklapp term for the doubly-counted states. This correction term is just

$$\lambda_c = \frac{\bar{V}^2(q) \Omega N(E_F)}{4v_s^2 MN} (1 - \cos \phi_{\max}),$$

or, putting in $V(q) = V(0)$ and simplifying,

$$\lambda_c = \frac{q_D^2 - (s - k_F)^2}{8sk_F}.$$

Now a reasonable choice for Π_u is $2k_F V(2k_F)$. With a Thomas - Fermi screened $V(q) \approx 1/(q^2 + k_s^2)$, $V(2k_F) = 0.296 E_F$ for the potassium parameters $(2k_F/k_s)^2 = 1.24$. Then

$$\lambda = \lambda_n + \lambda_u - \lambda_c = \frac{1}{2} + \frac{9}{4} (.089) \frac{k_F}{s} \ln \frac{q_D}{q_{\min}} - \frac{q_D^2 - (s - k_F)^2}{8sk_F}.$$

A plot of this function for potassium is shown in Figure 2.2.

The inclusion of an umklapp term λ_u in the present work is an extension of the usual 1 - O.P.W. calculation of (2.1). From this modified calculation we conclude that obtaining anisotropy in λ does not require the use of a complicated phonon spectrum or band structure. A linear phonon spectrum and free - electron Fermi surface have given rise to an anisotropy of about 20%. (The actual size of the anisotropy is dependent on the choice made for Π_u , the average value of $qV(q)$ over the umklapp region; the choice

Figure 2.2.. λ as a function of the angle θ of the initial wavevector with the reciprocal lattice vector.

of a Thomas - Fermi form factor evaluated at $2k_F$ probably overestimates the true potassium form factor in this region).

We conclude further that what such anisotropy in λ does require generally is that the electron wavevectors $[|k - k'|]$ in the matrix element $M_{kk'}$ of (1.1)] should vary over the Fermi surface in a different way from the phonon frequencies $\omega_{kk'}$; for when the variations in these quantities are the same, their ratio is a constant and the calculation shows no anisotropy. The present model has used the division into normal and umklapp terms to effect such a difference in variation. However, it is reasonable to suppose that physically more accurate electron band and phonon models will be required to reproduce experimental results accurately from such calculations of λ .

III. Exact Calculation of λ in Aluminum

A. Discussion of Units

The face-centered cubic lattice of aluminum has a single lattice parameter $a=4.04$ Å. The reciprocal space lattice, one unit cell of which is illustrated in Fig. 3.1, is body-centered cubic with a lattice parameter of $2\pi/a = 1.55 \text{ Å}^{-1}$. It is convenient to express distances in reciprocal space in units of this lattice parameter rather than in inverse Angstroms; thus the distance ΓX in Fig. 3.1 is taken to be exactly 1. The quantity $2\pi/a$ will be referred to as the calculational unit of (wavevector) length in reciprocal space.

From this fundamental choice of a calculational unit, other convenient units for area, energy, velocity and frequency immediately suggest themselves. The unit of area in reciprocal space should clearly be the length-unit squared. For energy, choose an electron of unit wavevector to have a unit free-electron energy:

$$(\hbar^2/2m) (2\pi/a)^2 = \text{one electron energy unit.}$$

With this choice, the free-electron energy is simply equal to the wavevector squared. The units of the electron velocity,

$$v_k = (1/\hbar) |\nabla_k E(k)|,$$

should be derived from those of k and E ; putting in the appropriate calculational units for k and E we find that

Fig. 3.1 The Brillouin zone of aluminum. The coordinates of the minimum symmetry element vertices are given in calculational units defined by $2\pi/a = 1$, where $a = 4.04 \text{ \AA}$ is the lattice parameter of aluminum.

$$(\hbar/2m) (2\pi/a) = \text{one velocity unit.}$$

For the phonon frequencies, let us first choose a unit phonon energy equal to one electron energy-unit:

$$\hbar\omega = (\hbar^2/2m) (2\pi/a)^2 .$$

Then associate this unit phonon energy with a unit phonon frequency so that

$$(\hbar/2m) (2\pi/a)^2 = \text{one frequency unit.}$$

Now all quantities in the expression (1.9) for λ have convenient calculational units except the two initial constants M and Ω , the aluminum ionic mass and the primitive cell volume. To be consistent, the units of Ω should be derived from the units of k ; since Ω is direct-space volume its units become $(2\pi/a)^{-3}$. It is convenient to retain the factor of $1/M$ outside the equation as a constant; another factor of $2m_e$ also comes out when all the physical quantities are converted to dimensionless calculational form. Table 3.1 summarizes the units; the resulting equation for λ can be written as:

$$\lambda(k) = \frac{2m_e}{MN} \int \frac{dS_{k'}}{|\nabla_{k'} E_{k'}|} \sum_{\sigma} \Lambda_{kk'}^{\sigma} , \quad (3.1)$$

with

$$\Lambda_{kk'}^{\sigma} = \left| \sum_{i,j=1}^4 \alpha_i \beta_j \hat{e}_{kk'} \cdot (k - G_i - k' + G_j) V(k - G_i - k' + G_j) \right|^2 / (\omega_{kk'}^{\sigma})^2 ,$$

where all quantities are in computational units.

21w

Quantity	Physical Units	Equation in Physical Units (or symbol)	Computational Units	Equation in Computational Units
1. wavevector	cm ⁻¹	(k)	2π/a	(k)
2. Fermi surface area element	cm ⁻²	(dS _k)	(2π/a) ²	(dS _k)
3. free-electron energy	erg	E _k = (ħ ² /2m)k ²	(ħ ² /2m) (2π/a) ²	E _k = k ²
4. electron velocity	cm/sec	v _k = (1/ħ) ∇ _k E(k)	(ħ/2m) (2π/a)	v _k = (1/ħ) ∇ _k E(k)
5. phonon frequency	sec ⁻¹	(ω _q ^σ)	(ħ/2m) (2π/a) ²	(ω _q ^σ)
6. ionic mass	g	(M)	m _e	(M)
7. unit cell volume	cm ³	Ω = a ³	(a/2π) ³	Ω = 8π ³
8. mass enhancement factor	dimensionless	(λ) *	dimensionless	(λ) †

=====

$$* \lambda_{\text{phys}} = \frac{g}{4\pi^3} \int \frac{dS}{|\nabla E|} \sum_{\sigma} \frac{|M_{kk'}^{\sigma}|^2}{\hbar \omega_{kk'}^{\sigma}}, \text{ where } M_{kk'}^{\sigma} = i \sqrt{\frac{\hbar}{2MN\omega_{kk'}^{\sigma}}} \sum_{j,l} \alpha_i \beta_j \epsilon_{kk'}^{\sigma} \cdot (k-k'-G_1+G_j) V(k-k'-G_1+G_j).$$

$$† \lambda_{\text{calc}} = \frac{2m_e}{MN} \int \frac{dS_{k'}}{|\nabla E|} \sum_{\sigma} \Lambda_{kk'}^{\sigma}, \text{ where } \Lambda_{kk'}^{\sigma} = \frac{1}{(\omega_{kk'}^{\sigma})^2} \left| \sum_{j,l} \alpha_i \beta_j \epsilon_{kk'}^{\sigma} \cdot (k-k'-G_1+G_j) V(k-k'-G_1+G_j) \right|^2$$

Table 3.1. Summary of physical and computational units and equations for various quantities. The final λ equations are for a 4 O.P.W. model.

B. Computer Implementation

An exact analytic calculation for λ cannot be carried out, since the phonon modes do not have closed-form solutions and since the Fermi surface itself is not accessible to exact analytic integration. The computer program which carries out the surface integral instead is conceptually very simple (see Fig. 3.2): a lot of points are stored that define the Fermi surface; the integrand is evaluated at each of these points and added to a running sum which becomes the total integral.

Actually, the points stored for the Fermi surface define only the $1/48$ minimum-symmetry-element (see Fig. 3.1); the 48 points equivalent to a given one are easily generated when required by the 48 reflections and rotations of the cubic group. In addition to coordinate data on the surface, it is convenient also to store the following quantities associated with each point:

- (1) the area element, dS ;
- (2) the electron velocity, $|v_k|$;
- (3) the four mixing coefficients for the electron state.

These quantities are invariant under the 48-fold lattice symmetry operation; even the ordering of the four mixing coefficients is retained, although the coordinates of the lattice vectors associated with them are transformed by lattice symmetry operations.

- Fig. 3.2 Flowchart of the surface integral program X-40PW.

In the following sections, each element in the calculation is discussed in detail; the analytic form of each quantity is given (where possible) and appropriate programming considerations are mentioned. It is worth mentioning here that the word "data" will generally refer to the computer-generated quantities outlined above; when experimentally derived data are meant instead, the word "experimental" will always be used.

1. The Fermi Surface Coordinate Data

a. Calculational Routine

The basic method used to obtain coordinates of points on the Fermi surface is the method described by Ashcroft⁷; the electron states are described by a 4 - orthogonalized - plane - wave model, and the corresponding secular equation implicitly defines all points that lie on the surface. Following Ashcroft, the secular equation is reduced for convenience of calculation to the form:

$$\begin{aligned} (T_0 T_3 - V_{200}^2) (T_1 T_2 - V_{200}^2) \\ - V_{111}^2 (T_0 + T_3 - 2V_{200}) (T_1 + T_2 - 2V_{200}) = 0, \end{aligned} \quad (3.2)$$

where

$$\begin{aligned} T_0 &= k^2 - E_k, \\ T_1 &= (k - \{1, 1, 1\})^2 - E_k, \\ T_2 &= (k - \{1, 1, \bar{1}\})^2 - E_k, \quad \text{and} \\ T_3 &= (k - \{2, 0, 0\})^2 - E_k. \end{aligned} \quad (3.3)$$

For points on the Fermi surface, we set $E_k = E_F$, the (constant) Fermi energy. In the calculational units used throughout the present work,

$$\begin{aligned} E_F &= 1.264, \\ V_{111} &= 0.0264, \quad \text{and} \\ V_{200} &= 0.0830. \end{aligned} \tag{3.4}$$

These are the values empirically determined by Ashcroft's Fermi surface fit ⁷.

The calculational routine to locate Fermi surface points then consists of locating points along specified line segments for which (3.2) is satisfied. The program which does this (FSEC-XYZ, Appendix A) operates efficiently by dividing a given line segment into n equal intervals and evaluating the left-hand side of (3.2) at the ends of each interval. Whenever the sign of the expression changes across an interval, it means that a root of the equation has been bracketed. At this point the program enters an iterated loop which brackets the required root within successively smaller limits. The operation of this iterative loop is illustrated in Fig. 3.3; it is often called the method of Regula Falsi⁸.

There are two possibilities of failure for this routine. The first is that the initial scan over n intervals may be too coarse, and two roots will be contained in an interval. If this happens, the sign of the polynomial will not be seen to change and the two roots will go undetected. This problem is solved easily by increasing the number of intervals. The second possibility is that a root will be double; in this case

ALGORITHM

- (1) Calculate $x' = x_1 + \frac{x_2 - x_1}{f_1 - f_2} \cdot f_1$. This is the crossing point of the straight line through (x_1, f_1) and (x_2, f_2) .
- (2) Evaluate $f' = f(x')$.
- (3) If $|f'| < f_0$, where f_0 is a predefined tolerance, then the process is considered finished and x' is taken to be the required root.
- (4) Otherwise, replace either (x_1, f_1) or (x_2, f_2) with the new point (x', f') so that the new interval again has end-values f_1, f_2 of opposite sign. Then go back to step (1).

Fig. 3.3 The iteration scheme used to locate roots.

the routine fails totally and the required point must be obtained by other means. It happens that there is indeed such a point in the aluminum Fermi surface, fortunately only one. This point was determined by a modified version of FSEC-XYZ that allows the user to bracket the desired point by interactive trial-and-error.

The data generated by the program were written into BASIC numeric random-access files ⁹ in order to avoid the truncation errors involved in converting the data to printed decimals and then (later, during a program run) back into binary-coded exponential form. Using random-access files also minimizes the amount of storage required.

b. Coordinate Systems and Data Storage

In principle one could simply store a single, exhaustive linear list of x,y,z coordinates to specify the Fermi surface. However, it is convenient at times to know which points are the neighbors of a given point; and it is efficient not to store redundant information. For these reasons the data are stored in compact form in several sets of files, with some coordinate a constant within a given file and changing by small increments from one file to the next in a given set of files. Then when the files are accessed by a running program the stored data and the file coordinate constants are used to generate full Cartesian coordinates in 3-space for each point.

The three sets of files used to define the second zone of the aluminum Fermi surface are shown in Fig. 3.4. This pattern of patches on the Fermi surface is by no means the simplest pattern, but it allows for varying point density on the surface. The patches that define zone 3 are illustrated in two views in Figs. 3.5-3.6; zone 1 is, of course, full.

In principle one would prefer to let the integration mesh in such a problem vary with the integrand and be calculated at the time the program itself is executed. However, it was found to be quite difficult in practice to generate reliable Fermi surface data without some human assistance to piece regions of the surface together. Thus, a fixed set of data files was used for evaluating $\lambda(k)$ at all points k . Fortunately, it was possible to define a few regions where the integrand variation was consistently large and others where it was consistently small, so that a finer mesh could be used where it was needed without making the calculation impractically long.

To explain the way the data are stored in files, it is necessary to introduce a "scaffolding" in space outside the Fermi surface. An example of such a scaffolding section is the plane defined by points Γ , A, and B in Fig. 3.4. All the points in the data-file labelled V1-10 are contained in this particular scaffolding plane; each point on the Fermi surface was originally found by starting with a point on the line segment AB and finding its projection on the Fermi surface with respect to the origin Γ ; and every point in the V1-10 section

is later determined by reading from the file V1-10 a single quantity r and transforming it back into x, y, z coordinates. If C is the point on the scaffold line AB and S is the point on the Fermi surface, then r is defined by:

$$r = \overline{FS}/\overline{FC}. \quad (3.5)$$

All that is left is the determination of the point C ; now in each file there is stored some number $n+1$ of r -values, and to these correspond $n+1$ points which are equally spaced along the scaffold line, beginning with point A in the example cited and ending with B . Thus, for the j th datum read from the file, the coordinates for the scaffolding point C are given by:

$$\begin{aligned} x_C &= x_A + j(x_B - x_A)/n, \\ y_C &= y_A + j(y_B - y_A)/n, \text{ and} \\ z_C &= z_A + j(z_B - z_A)/n, \end{aligned} \quad (3.6)$$

where j is understood to range from 0 to n .

Each data-file then contains a header of the following data: x, y, z coordinates of scaffold point #1; x, y, z coordinates of scaffold point #2; and n , the number of intervals to be constructed on the scaffold line (or, one less than the number of Fermi surface points stored). After that follow the $n+1$ values of r , and then the file is exhausted.

For the sections in the V1- and X- regions, the scaffold lines have been chosen to lie in the plane $k_x = 1$ and to be

parallel to the edge UW of the Brillouin zone. For the sections in the KL- region, the sections are determined by the requirement that adjacent regions should exactly touch. However, a further complication has been introduced into the file storage conventions for the KL- region. It was found from an earlier set of data - files that variation in the integrand for lambda is much greater near $k_z = 0$ than near L; thus the data points should be closer together near $k_z = 0$ than near L. Furthermore, a linear interpolation of points along the line PK (see Fig. 3.4) would lead to a rather non-uniform density of points in the $k_z = 0$ section of the Fermi surface. In fact, the curvature of the surface would cause points to cluster in the interior of this region, where they are least needed. For these reasons, the interpolation on the scaffold was generalized to a quadratic form. This allows the density on the scaffold to change uniformly from one end of the scaffold line to the other.

It was convenient to specify the parameter of this general interpolation to be the ratio h of the point densities at the two ends of the scaffold line; the functional form was then totally determined by requiring that the interpolation map the integers $[0,n]$ monotonically onto the real segment $[0,1]$. Then, for the j th datum read from a file, the equations for the scaffold point F on the scaffold line DE (see Fig. 3.4) are:

$$\begin{aligned} x_F &= x_D + s(x_E - x_D), \\ y_F &= y_D + s(y_E - y_D), \text{ and} \end{aligned} \quad (3.7)$$

$$z_F = z_D + s(z_E - z_D),$$

where

$$s = j(aj+b),$$

$$a = (1-h)/[n^2(1+h)], \text{ and} \quad (3.8)$$

$$b = 2h/[n(1+h)]$$

and again j ranges from 0 to n . This simplifies to the linear interpolation (3.6) when $h=1$, for then the quadratic factor a vanishes.

Exactly the same interpolation form was used to construct the scaffolding endpoints such as D and E (Fig. 3.4); these were considered as interior points along the lines PK and RL and generated at first by quadratic interpolation on those segments. After that, they were always retained as file constants stored in the header data in each file. The choice of a quadratic parameter h to generate points on PK and RL was governed by the requirement that intervals on the basal section of the Fermi surface be roughly equidistant; the value 0.55 was chosen finally by trial and error. Then the choice of interpolation parameter within each vertical section was guided by conflicting requirements: that the initial k increment in each file be around 0.0125, and that the resulting grid units be at least rhombic if not square. The final choices for h in these sections are 0.6 on PR and 0.3 on KL; intermediate values were calculated (arbitrarily) as quadratic interpolations themselves between the endpoint values.

Finally, the scaffolding for zone 3 must be described. Here it was possible to use the linear interpolation defined by (3.6), but since the zone 3 arm is not centered about the origin at Γ it is not possible to define the surface by projection through that point. Instead of a point Γ , lines GH and HJ (see Figs. 3.5, 3.6, 3.7) were used as centers for projection. Each file in zone 3 then had to have header data defining four rather than two scaffold points. The r-values stored in a file are defined for zone 3 by [cf. (3.5)]:

$$r = \overline{IS}/\overline{IM}, \quad (3.9)$$

where I is an interior point on GH (or HJ), S is the point on the Fermi surface, and M is the interior point on the outer scaffold line. Both I and M are determined by linear interpolation within their respective line segments; the equations for coordinates of I and M are just (3.6) with endpoints A and B chosen properly. Figures 3.5 and 3.6 show examples of scaffolding lines and points for zone 3.

Fig. 3.5 Patches defining the third zone of the Fermi surface in aluminum. Of the four patches, three are visible in this view; they are labelled W, V1, and X. Interfaces between patches are identifiable as changes in the density of data.

Fig. 3.6 Another view of the third-zone patches. The W-region is represented in this view by a coarser data grid than that used in the previous view.

2. Area Elements

The surface area associated with a point is calculated by finding the areas of the four parallelograms defined by the four nearest-neighbor points. The geometry is shown in Fig. 3.8. When a point is on a border of one of the patches on the surface, only the nearest neighbors that are inside the border are used; thus border points have roughly half, and corner points a quarter, of the area of interior points.

From Fig. 3.8 it seems possible that this method could be refined by including the next-nearest-neighbor points too, calculating areas for triangles rather than parallelograms. This refinement was not used because it would have involved numerous special cases to compute on all the possible border sites; further, in regions where such a refinement might be significant, other errors in the total calculation would probably dominate the result and make the improved areas worthless. In addition, a rough test of the two methods was made by constructing a grid on a spherical surface where an exact analytic expression could be used for the correct area element. From this, it seemed that the present parallelogram method was actually better, but that the two methods both gave areas with errors of much less than .01 % of the correct area.

The area element calculation was carried out by the program EDATA-V1, with modifications to suit each region of the

Fig. 3.8 The area element at P is found by constructing parallelograms with the nearest-neighbors A , B , C and D . This may be inaccurate if the next-nearest neighbors do not coincide nearly enough with the corresponding vertices of the parallelograms.

surface. See Appendix A for a list of this program and its modifications.

3. The Electron Velocity

The velocity defined as $v_k = (1/\hbar) |\nabla_k E(k)|$ could in principle be calculated as a vector derivative by the computer, i.e. by calculating $E(k)$ and $E(k+\Delta k)$ and multiplying the difference by $\Delta k/|\Delta k|^2$. Fortunately, this very inconvenient process is not necessary because an exact expression for v_k can be obtained.

Starting with the secular equation in the convenient form (3.2), take the (k -space) gradient of both sides:

$$\begin{aligned} & (T_0 T_3 - V_{200}^2) (T_1 \nabla T_2 + T_2 \nabla T_1) \\ & + (T_1 T_2 - V_{200}^2) (T_0 \nabla T_3 + T_3 \nabla T_0) \\ & - V_{111}^2 (T_0 + T_3 - V_{200}^2) (\nabla T_1 + \nabla T_2) \\ & - V_{111}^2 (T_1 + T_2 - V_{200}^2) (\nabla T_0 + \nabla T_3) = 0, \end{aligned} \quad (3.10)$$

where the T 's are defined already in (3.3). The expression (3.10) is linear in ∇E ; the solution can be written as:

$$\nabla E = 2N/D, \quad (3.11)$$

where

$$\begin{aligned}
N = & (T_0 T_3 - V_{200}^2) [T_1 (k - \{1, 1, 1\}) + T_2 (k - \{1, 1, 1\})] \\
& + (T_1 T_2 - V_{200}^2) [(T_0 + T_3)k - T_0 \{2, 0, 0\}] \\
& - V_{111}^2 (T_0 + T_3 - 2V_{200}) [2k - \{2, 0, 0\}] \\
& - V_{111}^2 (T_1 + T_2 - 2V_{200}) [2k - \{2, 0, 0\}], \quad (3.12)
\end{aligned}$$

and

$$\begin{aligned}
D = & (T_0 T_3 - V_{200}^2) (T_1 + T_2) + (T_1 T_2 - V_{200}^2) (T_0 + T_3) \\
& - 2V_{111}^2 (T_0 + T_1 + T_2 + T_3 - 4V_{200}). \quad (3.13)
\end{aligned}$$

It has been found that the denominator D becomes small, but does not vanish, at the numerically-determined contact point between zones 2 and 3; so the fraction N/D is apparently well-defined, at least for all wavevectors considered. Probably the values determined right at the contact point are nevertheless suspect, but their measure on the Fermi surface is very small; thus errors due to a bad velocity at such points are negligible.

The computer calculation for $|\nabla E|$ is implemented by breaking the numerator N into components that can be written as

$$\begin{aligned}
N_x &= X + D k_x, \\
N_y &= Y + D k_y, \text{ and} \\
N_z &= Z + D k_z, \quad (3.14)
\end{aligned}$$

where D is defined by (3.13) and

$$\begin{aligned}
X = & -(T_0 T_3 - V_{200}^2) (T_1 + T_2) - 2(T_1 T_2 - V_{200}^2) T_0 \\
& + 2V_{111}^2 (T_0 + T_1 + T_2 + T_3 - 4V_{200}), \quad (3.15.a)
\end{aligned}$$

$$Y = -(T_0 T_3 - V_{200}^2) (T_1 + T_2) + 2V_{111}^2 (T_0 + T_3 - 2V_{200}), \quad (3.15.b)$$

and

$$Z = (T_0 T_3 - V_{200}^2) (T_1 - T_2). \quad (3.15.c)$$

With these definitions, ∇E is simply

$$\nabla E = \{ 2(X/D + k_x), 2(Y/D + k_y), 2(Z/D + k_z) \}. \quad (3.16)$$

4. The Mixing Coefficients

The contributions to the electron state at k of the four plane-wave states are given by four mixing coefficients α_1 , α_2 , α_3 , and α_4 . These are obtained as eigenvector components of the matrix formed from the secular equation:

$$\begin{pmatrix} T_0 - E_k & V_{111} & V_{111} & V_{200} \\ V_{111} & T_1 - E_k & V_{200} & V_{111} \\ V_{111} & V_{200} & T_2 - E_k & V_{111} \\ V_{200} & V_{111} & V_{111} & T_3 - E_k \end{pmatrix} \cdot \hat{\ell} = v \hat{\ell} = 0, \quad (3.17)$$

where the T 's are defined above by (3.3). The subprogram EIGEN, given the secular matrix above, returns four eigenvalues v and associated eigenvectors $\hat{\ell}$; the program EDATA-V1 then searches through these four solutions for the one with v nearest zero and takes the corresponding $\hat{\ell}$ as the ordered set of mixing coefficients.

It was found that mixing coefficients could always be chosen by this method to yield a zero eigenvalue with a tolerance of 0.001.

5. Final Data-File Format

All the preceding data -- coordinates, area elements, velocities, and mixing coefficients -- are stored using the Dartmouth BASIC random access numeric file structure⁹. Using this format, a file containing data for n points will have a length in machine words of $L = 2 + h + 7n$, where n is the number of "header data" defining the file scaffolding. For files in zone 2, $h = 6$ or 7 ; for those in zone 3, $h = 12$. The total amount of storage required to define all the Fermi surface was 9691 words; an additional 1915 words was used to store alternate data with a different granularity for the X-region of zone 2 and the W-region of zone 3.

Throughout the data files, the data format is:

```
[ header data -- 6, 7, or 12 words ]
[ n, one less than the number of data pts. following]
[ point #0: coordinate r, area element, velocity,
    mixing coefficients]
[ point #1: . . . ]
. . .
[ point #n: . . . ]
```

6. Calculation of $V(q)$

The matrix element in (3.1) involves the lattice pseudo-potential form factor V evaluated at the phonon wavevector q . This quantity, along with the phonon frequencies and polarizations, cannot be stored in data-files ahead of time; it must be calculated anew for each phonon wavevector.

Ashcroft¹⁰ has reported a relatively simple formula for V which can be adjusted to give values for $V(111)$ and $V(200)$ that are nearly equal to those defined by the electron properties. In the computational units used here, the formula is

$$V(q) = -\frac{2}{3} E_F \frac{v^2 \cos(sx)}{x^2 + v^2 f(x)}, \quad (3.18)$$

where

$$x = q/(2k_F),$$

$$v^2 = 1/(\pi a_0 k_F),$$

$$s = 2k_F R_C,$$

$$f(x) = \frac{1}{2} + \frac{1-x^2}{4x} \ln \left| \frac{1+x}{1-x} \right| g(x), \quad \text{and}$$

$$g(x) = \frac{1-x^2}{1-2v^2 + 2x^2}.$$

Here a_0 is the Bohr radius (0.52917 Å) and R_C , interpreted physically as an ionic core radius, is the single parameter that can be adjusted to yield good values for $V(111)$ and $V(200)$. While $R_C = 0.59$ Å was given by Ashcroft as the value determined by the known principal band gaps (V_{111} and V_{200}), the value $R_C = 0.61$ Å was used in the present work because it reproduced V_{111} exactly and gave V_{200} with an error of 13%. (The choice $R_C = 0.59$ Å, by comparison, yielded errors in V_{111}

and V_{200} of -75% and -9%.) Figure 3.9 shows the form of $V(q)$ and also indicates the differences due to the two choices mentioned for R_c . Further discussion on choices for $V(q)$ is contained in Appendix B.

7. The Phonon Quantities ω_q^σ and $\hat{e}_q^\sigma$ Theory.

Given initial and final electron states k, k' , a corresponding phonon wavevector q is just $q = k - k'$. It is desired to find the frequencies ω and polarization vectors $\hat{e}$ for this wavevector q . There will be three values for each, corresponding to three phonon modes in aluminum, and these three choices are specified by the index $\sigma = 1, 2$, or 3 . It is well known¹¹ that these quantities can be calculated by solving the normal mode equation

$$D \hat{e}^\sigma = (\omega^\sigma)^2 \hat{e}^\sigma.$$

D is the dynamical matrix defined by

$$D_{ij}(q) = \frac{1}{M} \sum_{\ell} [1 - \cos(q \cdot R_\ell)] \frac{\partial^2}{\partial R_\ell^i \partial R_\ell^j} V(R) \Big|_{R=R_\ell},$$

where ℓ denotes a "shell" of atoms at the same distance from the original atom, and where the real-space lattice potential $V(R)$ is assumed to be a central-force potential. The sum is truncated in practice to a small number of shells. Hartmann and Milbrodt¹² report sets of force constants $\partial^2 V / \partial R_\ell^i \partial R_\ell^j$ for 8-shell and 21-shell models in aluminum, derived from experi-

mental phonon dispersion data.

Implementation

The present work has relied heavily on the implementation of the above theory contained in the Oak Ridge program GNU¹³. The sections of this FORTRAN program dealing with the definition and diagonalization of the dynamical matrix were extracted and translated into BASIC. In addition, the matrix definition routine was recast into a form that used a minimal number of subscripted variables. This reduced the final run-time of the program X-4OPW by about 20%, since it removed many of the very time-consuming boundary checks that are made in BASIC on a subscript each time it is referenced. The resulting block of code (subprogram ELEM in the file X4OPWLIB) is quite unwieldy.

Another run-time economy was made by using the 8-shell phonon model of the original GNU rather than the 21-shell model provided by Hartmann and Milbrodt¹⁴. Comparison of the resulting phonon spectra (see Fig. 3.10) indicated some minor differences in the large-q frequencies and very slightly lower frequencies for the 8-shell model in the low-q region. While no check was carried out on the polarization vectors, contour plots were made of the λ integrand for several regions of the Fermi surface (see Appendix B), and these exhibited only minor changes when the 8-shell model was substituted. Since the smaller model required roughly half as much execution time and reduced the total run-time by about 10%, it was used.

Figure 3.10. Phonon spectra calculated from two force-constant models.
 o - 21-shell model (78 force constants)
 + - 8-shell model (26 force constants)

8. The Matrix Element and Lattice Symmetries

Quite apart from the calculations discussed so far, there are some difficulties involved in calculating the matrix element defined by

$$M_{kk'}^{\sigma} = i \sqrt{\frac{\hbar}{2MN\omega_{\sigma}}} \sum_{i,j=1}^4 \alpha_i \beta_j \hat{e}_q^{\sigma} (k - G_i - k' + G_j) V(k - G_i - k' + G_j) \quad (3.19)$$

which is the matrix element between electron states in the formula (3.1). The difficulties arise from the way the calculation over the entire Fermi surface is carried out using symmetry operations on the minimum symmetry element of the surface.

Suppose that the point k at which λ is to be calculated lies somewhere on the symmetry plane $k_x = k_y$, e.g. the point P in Fig. 3.11. In the above sum, there will be a single term whose wavevector part $k - G_i - k' + G_j$ will reduce to just $k - k' + \{200\}$. Now, when the minimum symmetry element is reflected in the plane $k_x = k_y$, the corresponding form for (3.19) is exactly the same except k' and G_j are replaced by $F(k')$ and $F(G_j)$, where $F()$ is the appropriate symmetry operation. When this new form is used, the wavevector part of the selected term becomes $k - F(k') + \{020\}$. This is, in general, not equal to the old value for this term; hence the two contributions to the surface integral, arising from two symmetry-equivalent sections, cannot be expected to be equal. But clearly, if P lies on the symmetry plane, the two

Fig. 3.11 An initial wavevector p on the symmetry plane PLK . For such initial points it is incorrect to carry out the surface integral over the regions generated by reflection through the symmetry plane that contains the initial point.

contributions should be equal. The problem is that truncation of the model to four lattice vectors does not preserve the cubic symmetry for electron states; a state on the plane $k_x = k_y$, for example, should have equal (200) and (020) wavevector components, but in a 4-OPW model the second of these is always zero.

The way around this problem is simply not to calculate the matrix elements (3.19) for regions generated by reflection in a symmetry plane containing the initial point. Instead, the surface integral is evaluated for only the unreflected half of the surface. At three points of higher symmetry, namely L, K, and X, there are two symmetry planes involved and the surface integral is calculated for respectively $1/6$, $1/4$, and $1/8$ of the total surface. This necessary reduction to a symmetry region has the happy consequence of cutting program run times by the same fractions.

There is another question of lattice symmetry that must be discussed. This one concerns the way the third zone of the Fermi surface is represented. Actually, the two zones meet at the singularity called the "contact point." Yet the data base used in this work (see Fig. 3.4) represents zone 3 in an extended zone scheme that remaps the contact point so that no actual contact is evident. The question that arises from this situation is whether the set of phonons q generated in the course of a given calculation of λ is the same set that would be generated if zone 3 were represented like zone 2 in a

Figure 3.12. Remapping of the third-zone arm, showing a low- q phonon process not obviously described by the third-zone data as it is represented.

reduced zone scheme. Equivalently, how does an extended zone scheme ever "count" low- q processes such as the one sketched in Fig. 3.12?

The answer is that the set of phonons is not the same, but the set of matrix elements (3.19) is the same, and so is the value of λ . To see this, note that the third-zone arm that does make contact with zone two in the minimum symmetry element can be obtained from the original zone 3 data by a cubic group symmetry operation plus a lattice translation:

$$(\text{Remapped zone 3}) = \{111\} + R_{111} \mathcal{T}_{312} \cdot (\text{original zone 3}).$$

Now, to include such points in the calculation would mean forming matrix elements (3.19) with arguments $k - G_i - \mathcal{F}(k') + \mathcal{F}(G_j)$, where the operation $\mathcal{F}$ is now construed to indicate the cubic symmetry operation plus a lattice translation. But the lattice translation does not alter the argument, since it cancels out in the two terms of opposite sign. Thus the remapped arm that makes contact with zone 2 is equivalent to one already included in the calculation. The set of phonons is the same except for possible lattice vectors, which leave the phonon quantities ω_q^r and $\hat{e}_q^r$ invariant (see subprogram ELEM in the file X4OPWLIB, Appendix A, for this invariance). Thus all quantities in the formula for λ are unaffected by a lattice vector translation and the extended zone scheme really does work.

C. Results

1. Values for λ

A summary of all the values calculated for λ is presented in Tables 3.2 and 3.3. The results of Tables 3.2 and 3.3 are plotted also in Figures 3.15 - 3.17. The horizontal scale for these plots is the arclength along the appropriate section of the Fermi surface, as calculated from the Fermi surface data used throughout. The behavior of λ is discussed below in section 3.

2. Comparison with Experimental Data

There are no experiments to measure λ at individual points. However, the line-integral average of λ around an orbit should equal the cyclotron mass-enhancement factor:

$$\frac{m^*}{m} = \frac{1}{\pi} \oint \frac{dl (1+\lambda)}{\hbar v_k} \quad (3.20)$$

This quantity is measured in de Haas-van Alphen effect and Azbel'-Kaner resonance experiments. Ashcroft¹⁵ has collected experimental results from various sources and compared them with values obtained by machine calculation from the formula:

$$\frac{m^*}{m} = \frac{1}{\pi} \cdot \frac{dA}{dE} \quad (3.21)$$

This formula just represents an unenhanced (band) mass, one that does not include effects of the phonon spectrum; it is

datum *	coordinates k_x, k_y, k_z	location†	λ	comments
1	.5835, .5835, 0.	3 19 0	0.5310	nearest K on Brillouin zone
2	.5821, .5821, .01009	3 19 1	0.5264	
3	.5483, .5483, .06127	3 19 5	0.4533	
4	.4790, .4790, .1442	3 19 10	0.3929	midway from K to L
5	.3497, .3497, .3497	3 19 19	0.3676	nearest L on B.z.
6	.5686, .2636, .2636	3 10 19	0.3823	
7	.7320, .2253, .2253	3 4 19	0.4135	
8	.8267, .2112, .2112	3 1 19	0.4421	
9	.8873, .2033, .2033	1 2 0	0.4939	on " V_{111} " ridge
10	.8949, .1243, .1243	2 2 0	0.4401	using fine data grid at X
11	.8863, .07696, .07696	2 3 0	0.4293	using coarse X data
12	.8810, 0., 0.	2 6 0	0.4244	coarse X data
13	.8919, .1549, 0.	2 3 3	0.4327	coarse X data
14	.9516, .3814, 0.	1 7 34	0.4715	
15	.9582, .3964, 0.	1 6 35	0.4743	points 14,15,16 on " V_{111} " ridge
16	.9464, .4134, 0.	1 4 37	0.5743	
17	.8952, .4203, 0.	1 1 40	0.5674	
18	.8613, .4275, 0.	3 1 0	0.5626	
19	.7133, .4854, 0.	3 10 0	0.5405	on " V_{200} " ridge
20	.5844, .5560, .03381	3 18 3	0.4558	
21	.5786, .5214, .0605	3 17 5	0.4553	
22	.5627, .4448, .1176	3 15 9	0.4077	
23	.5460, .3204, .2230	3 12 16	0.3786	

* See Fig. 3.13 below for a plot of the locations of these points.
† The numbers index: file region, section #, datum #.

Table 3.2 Values obtained for λ on zone 2 of the Fermi surface.

Figure 3.13 Locations of points on zone 2 where λ has been calculated.

datum *	coordinates k_x, k_y, k_z	location†	λ	comments
1	.7012, .7012, 0.	7 0 0	0.3710	nearest K on B.z.
2	.7046, .7046, .0127	7 1 0	0.3761	
3	.7359, .7359, .0425	7 4 0	0.3971	points 1-9 on [110] plane
4	.7606, .7606, .0560	4 0 0	0.3895	
5	.7792, .7792, .0610	4 2 0	0.3623	
6	.7884, .7884, .0547	4 4 0	0.3526	on "V ₁₁₁ " ridge
7	.7908, .7908, .0471	4 5 0	0.3627	
8	.7932, .7932, .0272	6 0 0	0.3820	
9	.7940, .7940, 0.	6 2 0	0.3899	
10	.8398, .7472, 0.	6 2 5	0.3938	
11	.9177, .6550, 0.	5 0 0	0.4195	all points in file region
12	.9537, .6115, 0.	5 0 6	0.4575	5 using fine data grid
13	.9763, .5940, 0.	5 0 9	0.5158	for the neck
14	.9923, .5895, 0.	5 0 11	0.5664	
15	1.000, .5891, 0.	5 0 17	0.5752	
16	1.000, .5884, .0277	5 3 12	0.5754	points 16 & 17 are adjacent to contact point.
17	1.000, .5843, .0325	5 5 12	0.4768	nearest W on B.z.
18	1.000, .5448, 0.	5 12 12	0.4746	
19	.9763, .5926, .0140	5 3 9	0.5071	
20	.9760, .5893, .0207	5 6 9	0.4864	
21	.9729, .5689, .0134	5 8 9	0.4531	
22	.9721, .5618, 0.	5 12 9	0.4501	

* See Fig. 3.14 below for a plot of the locations of these points.

† The numbers index: file region, section I, datum #.

Table 3.3 Values obtained for λ on zone 3 of the Fermi surface.

Figure 3.14 Locations of points on zone 3 where λ has been calculated.

convenient to refer to (3.21) as the formula for the band effective mass, and to (3.20) as that for the quasiparticle effective mass. It is possible, in addition, to calculate a band effective mass by performing the line integral as in (3.20) but without taking λ into account; so a third formula, equivalent to (3.21), is:

$$\frac{m^*}{m} = \frac{1}{\pi} \oint \frac{dl}{\hbar v_k} . \quad (3.22)$$

The values obtained from (3.22) provide a convenient check against the values obtained by Ashcroft from (3.21).

The results for m^*/m using (3.20) are presented in Table 3.4 together with experimental values and the Ashcroft band values.

3. Discussion of variation in λ

The behavior of λ is consistent with the approximations made in Chapter II for jellium and the 1QEW potassium model. In free-electron-like regions of the Fermi surface λ is very close to the approximate jellium value of 0.5.

The curious thing about λ is the way variations in multiple-QEW regions seem to depend on the zone index (2 or 3). On zone 3, departures from free-electron behavior are associated with a smaller λ rather than a larger one as on zone 2. This would seem to depend on the way the mixing-coefficient four-vectors project onto one another; certainly it is not easy

Fig. 3.15 Variation in λ along the second-zone 110 section. The letters K, L, U, and X indicate places along the section that are closest to the vertices of the Brillouin zone.

Fig. 3.16 Variation in λ along the second-zone 111 and 001 sections. The point labelled M in the 111 section was obtained by interpolation from the nearby points at which λ was calculated. Interior points along the 111 section correspond to nearest grid points, not on the 111 plane, so the λ values are not accurate.

Fig. 3.17 Variation in λ along the third-zone 110, "150", and 001 sections. The "150" section approximates the extremal-area neck section. The 001 section corresponds to the inner ring orbit.

to imagine such an effect to be due to anisotropy in the phonon spectrum. Nor can it be accounted for plausibly as a simple geometrical umklapp effect. This variation in multiple-OPW regions is a clear indication that correct calculations of λ do require inclusion of the correct electron states, i.e. of the mixing coefficients.

4. Effective Masses

Our quasiparticle values for m^*/m are in very good agreement with experiment, considering that values of V_{111} and V_{200} are only determined to about 10%. However, the band masses calculated from (3.22) did not agree with Ashcroft's exactly. In principle we should expect identity between the two calculations for m^*/m , i.e.

$$\oint \frac{d\ell}{\hbar v} = \frac{dA}{dE}.$$

The discrepancy can be attributed to the fact that the V 's used to determine the Fermi surface are actually energy dependent, not constants as assumed in the present work. Ashcroft did make a correction for this energy dependence in calculating the derivative dA/dE . In the present work, this dependence enters through the electron velocity v_k ; the correction involves additional terms in (3.10) that make explicit the dependence of V_{111} and V_{200} on E , so that the vV terms are not zero as assumed.

1.	2.	3.	4.	5.	6.	7.
Orbit	Experimental Cyclotron Mass ¹	Ashcroft Band Mass ²	Average Enhanced Mass ³	Quasiparticle Enhanced Mass ⁴	Orbital Effective λ ⁵	Uncorrected Band Mass ⁶
zone 2						
110	1.20	0.871	1.30	1.24	0.42	0.816
111	1.29	0.95	1.42	1.35	0.42	
zone 3						
110	0.130	0.095	0.142	0.131	.38	0.0896
001	0.73	0.495	0.738	0.712	0.44	0.504
"150"	0.091	0.0605	0.090	0.0894	0.48	0.0826

Notes

1. For sources of the experimental values, see Ashcroft (Ref. 15).
2. Calculated from $m^*/m = (1/\pi) dA/dE$.
3. Column 3 result times the average specific-heat λ of 0.49 for aluminum given in N. W. Ashcroft and J. W. Wilkins, Phys. Letters 14, 285 (1965).
4. Calculated from $m^*/m = (1/\pi) \oint dk(1+\lambda)/\hbar v_k$. This result includes the energy dependence of the pseudopotential. But see column 7.
5. Column 5 result divided by column 3 result, minus 1.
6. Calculated from $m^*/m = (1/\pi) \oint dk/\hbar v_k$, but uncorrected for the energy dependence of v_k (see text).

Table 3.4. Calculated values of λ , compared with band values calculated previously by Ashcroft, and with experimental values.

Thus it seems that the results for λ have to be scaled by a correction factor to take into account the energy dependence of the V 's. This was done for the effective masses (not for the λ 's) by multiplying the quasiparticle m^*/m by the ratio of the Ashcroft band mass to the line integral band mass (3.3.22). The final values for the quasiparticle masses given in Table 3.4 are these corrected values.

IV. Quasiparticle Lifetime Calculations in Aluminum

A. Formulas for τ^{-1}

The inverse lifetime for a single quasiparticle excitation on the Fermi surface, at temperature T , is given by^{2,16,17}:

$$\tau^{-1} = \frac{\Omega}{4\pi^3} \int \frac{\partial S_{k'}}{|\mathbf{v}_{k'} E_{k'}|} \sum_{\sigma} \Lambda_{kk'}^{\sigma} F(\omega^{\sigma}, T), \quad (4.1)$$

where Λ^{σ} is defined in (3.1) and where

$$F(\omega, T) = \omega / [(e^{\hbar\omega/kT} + 1) (1 - e^{-\hbar\omega/kT})]. \quad (4.2)$$

Since (4.1) is just the equation for λ with the extra factor $F(\omega, T)$ in the integrand, the computer program that calculates λ can be modified to give the temperature - dependent τ^{-1} as well. Instead of accumulating a single integrand, this modified program (T-4OPW, Appendix A) accumulates a list of integrands at various temperatures. The units of the extra factor of ω are adjusted to physical units, i.e. sec^{-1} .

For temperatures near enough to $T=0$ K, it will be seen that $\tau^{-1} \propto T^3$. Thus it is reasonable to try to simplify (4.1) in the limit of low temperatures to find the constant of proportionality, the T^3 coefficient of τ^{-1} . (All the results of this chapter will be referred to this coefficient, i.e. to τ^{-1}/T^3 .) Such a simplified calculation has the advantage that the low-temperature form of the integrand in (4.1) vanishes except in the region around $q=0$; the size of the calculation is thus drastically reduced. Also, since the T^3 coefficient calculation is essentially independent of the full τ^{-1} calcu-

lation of (4.1), it provides a much-needed check on the results of (4.1).

To obtain a low-T limit for (4.1), write the surface integral as

$$\int dS_k = \int_0^{2\pi} d\phi \int_0^\infty q dq,$$

and take $\omega^\sigma = v^\sigma q$. Define $x^\sigma = \hbar\omega^\sigma/kT$. Then

$$\tau^{-1} = \frac{\Omega}{4\pi^3} \left(\frac{kT}{\hbar} \right)^3 \frac{1}{2MN|vE|} \left[\int_0^\infty \frac{x^2 dx}{(e^x+1)(1-e^{-x})} \right] \int_0^{2\pi} d\phi \sum_\sigma \frac{(\mathbf{T}^\sigma)^2}{(v^\sigma)^4}, \quad (4.3)$$

$$\text{where } \mathbf{T}^\sigma = \lim_{q \rightarrow 0} \frac{1}{q} \sum_{i,j} \alpha_i \beta_j \hat{\epsilon}^\sigma \cdot (\mathbf{k}-\mathbf{k}'-\mathbf{G}_i+\mathbf{G}_j) V(\mathbf{k}-\mathbf{k}'-\mathbf{G}_i+\mathbf{G}_j).$$

This limit can be written as a diagonal sum of O.P.W. terms for which $\sum_{i=1}^4 \alpha_i \beta_i = 1$, plus a sum of off-diagonal terms using a first-order Taylor expansion of β and V . The integral over x in (4.3) evaluates numerically to 2.103. The final result is:

$$\tau^{-1}/T^3 = \frac{2.103}{8\pi^3 MN|vE|} \left(\frac{k}{\hbar} \right)^3 \int_0^{2\pi} d\phi \sum_\sigma \frac{(\hat{\epsilon}^\sigma \cdot \mathbf{A})^2}{(v^\sigma)^4}, \quad (4.4)$$

$$\begin{aligned} \text{where } \mathbf{A} = & -\hat{\mathbf{q}}V(0) + \sum_{i=1}^3 \sum_{j=1}^4 \left\{ \left[\alpha_i \frac{\partial \beta_j}{\partial q} - \alpha_j \frac{\partial \beta_i}{\partial q} \right] V(\mathbf{G}_j - \mathbf{G}_i) \right. \\ & \left. - 2\alpha_i \alpha_j \frac{\partial V(\mathbf{G}_j - \mathbf{G}_i)}{\partial q} \frac{\hat{\mathbf{q}} \cdot (\mathbf{G}_j - \mathbf{G}_i)}{|\mathbf{G}_j - \mathbf{G}_i|} \right] (\mathbf{G}_j - \mathbf{G}_i) - \left[(\alpha_i \beta_j + \alpha_j \beta_i) V(\mathbf{G}_j - \mathbf{G}_i) \right] \hat{\mathbf{q}} \right\}. \end{aligned} \quad (4.5)$$

Now the calculation of τ^{-1}/T^3 from (4.4) depends on the same computer-generated electron and phonon quantities that (4.1) uses, but the integral in (4.4) only calls for an angular integration over nearby points. The program T3 (see Appendix A) carries out this angular integration by trapezoidal summation over points equally spaced along the nearest grid-lines that enclose the initial wavevector point $\mathbf{k}$. This

scheme is illustrated in Figure 4.1(a). Values of the mixing coefficients β are linearly interpolated from the tabulated values at the grid points, while the phonon quantities $\hat{\epsilon}$ and ω are calculated anew for each q .

Comparison of the results of (4.4) and (4.1) suggested that the surface integral program T-4OPW was not calculating τ^{-1} accurately at low temperatures. This was not unexpected, since the element in the integrand with $q=0$ is not added to the sum (the $q \rightarrow 0$ limit of the integrand is not well-defined; it depends on the direction of q). With the integrand peaking strongly at $q=0$ for low T , a significant part of the total integral was lost. The T-4OPW results were improved by calculating the $q=0$ element integrands separately and adding them to the previous surface integrals. The program to do this (TQ-0, Appendix A) must calculate an angular average similar to the one calculated by the T^3 coefficient program T3. Here the points to be averaged over are considered to lie on the border of the grid cell with center at $q=0$. The geometry is illustrated in Figure 4.1(b). Again, the mixing coefficients β along the cell border were found by linear interpolation from the tabulated grid-point values.

The form for this zeroth-element calculation is (cf. 4.1):

$$\tau_0^{-1} = \frac{\Omega}{4\pi^3 |v_E|} \left(\frac{kT}{h} \right)^3 \sum_{\sigma} \int_0^{2\pi} d\phi \frac{\Lambda^{\sigma}}{(v^{\sigma})^2} f(y_T^{\sigma}), \quad (4.6)$$

where $y_T^{\sigma} = \frac{h}{kT} \omega^{\sigma}$ for q on the cell boundary as in Figure 4.1(b). Here $f(y)$ is just the bracketed integral in (4.3) with a finite upper limit:

4.1(a). Grid Geometry for limiting low-T calculation of T^3 coefficient of τ^{-1} . Final states k' are interpolated linearly from grid states on the border; the example shown is interpolated from the two grid points 6 and 7.

4.1(b). Grid Geometry for calculating the $q=0$ integrand of τ^{-1} . Final states k' are interpolated linearly from the center grid point and the border points; the example shown is interpolated from the three points 0, 6 and 7.

Figure 4.1. Use of the Fermi surface data grid in the programs (a) T3 and (b) TQ-0.

$$f(y) = \int_0^y \frac{x^2 dx}{(e^x+1)(1-e^{-x})} \quad (4.7)$$

The value of τ_0^{-1}/T^3 from (4.6) is a monotonically decreasing function of temperature. It seems to be about 10% of the integral (4.1) at 5 K, and it falls into insignificance around 20 K.

The zeroth element formula (4.6) has an additional use in that it provides a check on the T^3 - coefficient calculation; in the limit as $T \rightarrow 0$, $f(y) \rightarrow 2.103$ and (4.6) becomes

$$\tau^{-1} = \frac{2.103}{4\pi^3 |\nabla E|} \left(\frac{kT}{h} \right)^3 \sum_{\sigma} \int_0^{2\pi} d\phi \frac{\Lambda^{\sigma}}{(v^{\sigma})^2}, \quad (4.8)$$

where now the entire integral for τ^{-1} is effectively contained in the zeroth element. The availability of this alternate calculation was of crucial importance in debugging the programs for τ^{-1}/T^3 . The final T^3 coefficients calculated from (4.4) and (4.8) agreed to within 1%. The source of this small discrepancy seems to be the different methods used to evaluate the form factor $V(q)$; in (4.8) values of $V(G+q)$ were evaluated directly using the Ashcroft form of this potential, while in (4.4) the first term of a Taylor series expansion of $V(G+q)$ about $q=0$ was used.

The final computations, then, were carried out at finite temperature using (4.1) augmented by the zeroth-element formula (4.6); and at zero temperature, using (4.4) with the alternate formula (4.8) to check on the results.

B. Results

The results for τ^{-1} using (4.1) and (4.6) are presented graphically in Figures 4.2-4.4 as log-log plots. The quantity displayed on the vertical scale is actually $\log (\tau^{-1} / T^3)$; thus flat places on the graphs are identified with temperatures for which $\tau^{-1} \propto T^3$. The results for the T^3 coefficient calculated from (4.4) [or (4.8)] are displayed on the same graphs, as asymptotes drawn in at 1 K (since values at 0 K cannot be shown on a log-log plot directly). These same results are presented numerically in Table 4.1.

Variation in τ^{-1} / T^3 in the $T \rightarrow 0$ limit along the [110] sections of zones 2 and 3 is shown in Figures 4.5 and 4.6. Where available, values of τ^{-1} / T^3 at 5 K and 10 K have been included in these figures. Figure 4.7 shows the behavior of τ^{-1} / T^3 along the "neck" section of zone 3; no calculations of τ^{-1} were made for $T > 0$ K along this section, however.

Figure 4.2. τ^{-1}/T^3 as a function of temperature on the 110 section of zone 2. The limiting low-T asymptotes are indicated at 1K. See Table 4.1 and Figure 3.13 for relation of points to Fermi surface. This figure shows the variation from K to L.

Figure 4.3. τ^{-1}/T^3 as a function of temperature on the 110 section of zone 2, from X to L. The limiting low-T asymptotes are indicated at 1K.

Figure 4.4. τ^{-1}/T^3 as a function of temperature on the 110 section of zone 3. The limiting low-T asymptotes are indicated at 1K. The points are identified by their locations in file region, section index, and datum index (see Fig. 3.14 and Table 3.3 for these). The two points labelled (6 2 0) and (7 4 0) are free - electron - like.

Temp. (°K)	Location						
=====	=====						
	3 19 0	3 19 1	3 19 2	3 19 3	3 19 5	3 19 10	3 19 19
0	3192800	2967300	2082000	1100200	346315	183262	177600
5	2306682.	2148323.	1782835.	1251827.	394693.	165035.	171944.
10	1739844.	1651957.	1428576.	1136704.	544870.	183174.	182726.
15	1494690.	1432547.	1262117.	1033811.	582132.	194759.	181998.
20	1430922.	1376593.	1222160.	1011493.	605663.	213276.	184873.
25	1406967.	1357813.	1215450.	1018364.	637849.	243613.	199132.
30	1384688.	1340608.	1211465.	1030477.	675949.	286330.	228355.
35	1359829.	1320647.	1204972.	1041269.	715122.	337344.	269779.
40	1334114.	1299389.	1196318.	1049335.	751881.	391167.	317664.
45	1308790.	1277952.	1186083.	1054313.	784060.	443206.	366562.
50	1284031.	1256533.	1174351.	1055969.	810508.	490420.	412583.
55	1259502.	1234824.	1160916.	1054121.		531152.	453422.
60	1234675.	1212383.	1145524.	1048699.		564750.	487981.
65	1209122.	1188853.	1128003.	1039757.		591219.	515967.
70	1182572.	1164031.	1108336.	1027492.		610940.	537579.
75	1154915.	1137863.	1086617.	1012206.		624520.	553294.
80	1126185.	1110424.	1063054.	994262.		633022.	563727.
85	1096510.	1081880.	1037911.	974065.		636086.	569538.
90	1066078.	1052445.	1011484.	952023.		635496.	571383.
95	1035107.	1022366.	984079.	928535.		631545.	569879.
100	1003824.	991880.	955997.	903965.		624825.	565590.

Table 4.1. Values of τ^{-1}/T^3 for various Fermi surface points at temperatures from 0 to 100 K. The 0K values are limiting T→0 values calculated by the program T3; the rest are calculated by the program T-4OPW.

The locations of the data points refer to the Fermi surface grid data; see Figs. 3.13, 3.14 and Tables 3.2, 3.3 for these locations.

continued+++

Temp. (°K)	Location					
=====	=====					
	1 5 0	2 10 0	7 0 0	7 4 0	4 4 0	6 2 0
0	2189600	186317	9843700	597834	2812900	206998
5	1509901.	170827.	6293140.	1196489.		432847.
10	1931926.	189090.	4062212.	2327710.	2315987.	1201285.
15	1860626.	232102.	3142560.	2601567.		1477392.
20	1735613.	332532.	2725276.	2544875.	1792714.	1445872.
25	1603192.	459596.	2465396.	2399736.		1343497.
30	1488457.	581138.	2258255.	2236010.	1518408.	1251048.
35	1399501.	682768.	2078885.	2077689.		1183375.
40	1334109.	761680.	1921120.	1932633.	1358134.	1136165.
45	1286518.	819811.	1782200.	1802428.		1102109.
50	1250847.	860410.	1659684.	1686233.	1244222.	1075267.
55	1222326.			1582351.		1051679.
60	1197509.			1488967.		1028976.
65	1174104.			1404433.		1005881.
70	1150688.			1327365.		981814.
75	1126492.			1256629.		956607.
80	1101168.			1190776.		930334.
85	1074655.			1130755.		903192.
90	1047056.			1074342.		875439.
95	1018562.			1021637.		847328.
100	989416.			972278.		819106.

Table 4.1. continued.

Figure 4.5. Anisotropic variation in τ^{-1}/T^3 along the [110] section of zone 2. The limiting low-T value is plotted as a continuous line with values at 5 K and 10 K shown by points.

Figure 4.6. τ^{-1}/T^3 along the [110] section of zone 3. See Fig. 3.14 and Table 3.3 for the locations of the points labelled "7 0 0" etc.

Figure 4.7. τ^{-1}/T^3 along the "neck" section of zone 3. See Fig. 3.14 and Table 3.3 for the locations of points labelled "5 0 9", etc. The points shown are all limiting low-T values.

1. Umklapp Effects

Referring to Figures 4.2 and 4.3, it is interesting that, for free - electron points near X and L, raising the temperature has no effect on τ^{-1}/T^3 until some critical temperature -- 10 K at X, and 20 K at L -- is reached. Then the behavior of τ^{-1} changes from T^3 to T^4 . It is plausible to identify this transition with the onset of "umklapp" effects, i.e., with the sudden increase in M_{kk} , due to inclusion of states located roughly one lattice vector away. What happens is that in calculating the integral (4.1) the effective region of integration around $q=0$ increases with temperature. Starting at a free - electron point like X or L, τ^{-1} goes as T^3 until this effective region begins to include other parts of the Fermi surface that are "nearby" under translations of the lattice. Then τ^{-1} begins to vary as T^4 . To illustrate that the T^4 behavior is indeed associated with the Fermi surface geometry, τ^{-1} versus T has been calculated with the integral restricted to the free - electron - like region of the Fermi surface around X (see Fig. 4.8). It is clear that the T^4 behavior has been suppressed. Then, allowing the integral to cover the nearby "V1-ridge", the value of τ^{-1}/T^3 remains the same for temperatures up to about 20 K and then starts to get larger than the value using just the region of the surface around $q=0$. By putting in all the Fermi surface, considerable deviation from the $q=0$ region result begins to occur by about 10 K. It is tempting to identify this effect at X with final states that lie on the other side of the second zone, i.e. on

Figure 4.8. Comparison of full τ^{-1}/T^3 calculation at X on zone 2 with the "reduced" calculation using only parts of the Fermi surface near X. The names "X" and "V1" are explained in Fig. 3.4. The limiting low-T value for τ^{-1}/T^3 is indicated by the line at 1K.

Figure 4.9. Comparison of full τ^{-1}/T^3 calculation at L on zone 2 with "reduced" calculation using only the part of the Fermi surface near L. The name "KL" is explained in Fig. 3.4. The limiting low-T value for τ^{-1}/T^3 is indicated by the line at 1K.

Figure 4.10. Comparison of full τ^{-1}/T^3 calculation at K on zone 2 with "reduced" calculation using only the part of the Fermi surface near K. The limiting low-T value for τ^{-1}/T^3 is indicated by the line at 1K.

the X-face in the $[1\ 0\ 0]$ direction; these states are actually rather close to the states at X when the lattice symmetry is invoked. Figure 4.9 is a similar plot at L on zone 2 which illustrates the same principle. Comparison with the full calculation shows how little the complete Fermi surface contributes to τ^{-1} at low temperatures. Figure 4.10 shows how a reduced calculation of the same sort looks at K on zone 2. It is significant to note that the reduced calculation, which only involves a small segment of the total Fermi surface and hence a small amount of computer time, is equivalent to the full calculation for temperatures below 10 K or so.

2. Regime of T^3 Behavior

Apparently the umklapp effect described above acts very near $T=0$ K to suppress T^3 behavior of τ^{-1} on the Fermi surface ridges. Thus, given the limited temperature resolution of the present work, it is difficult to identify a T^3 temperature regime for points near U and K on zone 2, or for the corresponding ridge points on zone 3. (Presumably there is such a regime for these points, very near $T=0$ K, but the use of the program T-40PW at low temperatures is suspect. This is discussed below under "Errors").

For free - electron - like points, T^3 regimes are easy to identify. At X on zone 2, τ^{-1}/T is constant from 0 K up to about 10 K. And at L on zone 2, the T^3 regime extends slightly farther, from 0 K up to about 20 K. On the $[110]$ section of zone 3, the two regions of more or less free -

electron - like surface already exhibit the umklapp T^4 behavior at 5 K, so the T^3 regime lies somewhere below this temperature.

3. The Low - T τ^{-1}/T^3 Anisotropy

Figures 4.5 - 4.7 show the results of calculating τ^{-1}/T^3 from (4.4) in the limit as $T \rightarrow 0$. The identification of peaks in τ^{-1}/T^3 with multiple - O.P.W. regions of the Fermi surface is quite clear, at least in Figures 4.5 and 4.6; in Figure 4.7 this identification is not so obvious because the neck section does not have any marked free - electron regions. From the points plotted at 5 K and 10 K, it is also clear how raising the temperature tends to wash out the peaks. On zone 2 (Fig. 4.5), raising the temperature to 5 K attenuates the peak at "K" to about 60% of its height at 0 K. At 10 K, the peak has not only been reduced to roughly half its value at 0 K, but the width of the peak has been increased by a factor of about 2. On zone 3 (Fig. 4.6), the washout is even more pronounced. By 10 K, the peak structure no longer dominates the plot. This quicker washout is entirely reasonable, considering that the third zone is much smaller than the second and thus shows more sensitivity to umklapp effects associated with an increase in the effective region of integration.

4. Errors in the τ^{-1} Calculation

The primary source of error in the calculation of τ^{-1} from (4.1) lies in the coarseness of the Fermi surface data grid. With the integrand varying quite rapidly near $q=0$ and dropping in effect to zero at distant points, the fixed data grid becomes almost absurd. Now, since the integral is simply a sum of integrands multiplied by area elements, the major error will be due to rapid integrand variation around the zeroth element which the data grid does not pick up. (It was observed that the integrand displayed strong angular variation around $q=0$ for "ridge" points). The lower the temperature, the worse this error becomes. Thus the data grid imposes a low - temperature limit on the accuracy of the calculation (4.1). [The low - T formula (4.4) is not invalidated by these remarks; for (4.4) includes an angular integration over a sub-grid constructed to take the angular variation into account].

To establish the limit on (4.1) it is helpful to try to match up the limiting low - T result from (4.4) with the temperature plots of (4.1). Such matching suggests that at the second - zone free - electron - like points X and L, τ^{-1} from (4.1) is correct at 5 K and above. Then it is reasonable to take 5 K as a lower limit everywhere on the validity of the τ^{-1} calculation via T-4OPW. (Some runs of T-4OPW for lower temperatures did show fluctuations in τ^{-1}/T^3 that seemed to be accountable to truncation in the integration routine).

The results at X and L, where angular variation in the integrand is not great, are probably accurate to within 5% or less; the agreement with the limiting low - T results is very good. But at other points, with O.P.W. ridges very close, angular variation in the integrand begins to exceed the ability of the data grid to follow it and truncation of the integral begins to be significant. Since the zeroth - element calculation of TQ-0 altered the results for τ^{-1} by about 10 %, it seems safe to use this same percentage as an estimate of the error in all the τ^{-1} values at ridge points.

Appendix A

This appendix contains lists of the programs used in the present work.

<u>Program name</u>	<u>Page</u>	<u>Comments</u>
FSEC-XYZ	82	Finds points on Fermi surface.
EDATA-V1	83	Calculates electron properties.
ED-ALTS	86	Alters to EDATA-V1 for other regions.
EIGEN	87	Calculates eigenvalues, vectors.
X-4OPW	89	Calculates 4-O.P.W. λ .
X4OPWLIB	92	Subprograms for X-4OPW.
FILESPEC	94	Specifies data files for X-4OPW to use.
FCONST1	95	21-shell phonon force constants.
FCONST2	95	8-shell phonon force constants.
V-ALTS	96	New and old versions of $V(q)$ routine.
T-4OPW	96	Calculates 4-O.P.W. τ^{-1} vs. T .
T3	100	Calculates low- T limit of τ^{-1}/T^3 ,
TQ-0	103	Calculates zeroth element of τ^{-1} vs. T for addition to T-4OPW result; also calculates low- T limit of τ^{-1}/T^3 for comparison with T3 result.

FSEC-XYZ

```

5 * FSEC-XYZ A B HEADOR 17.9.74 STORED ON TAPE
7 * LAST CHANGE 08/25/74 4-POINT INTERPOLATION IMPLEMENTED
8
50 PRINT "OUTPUT FILE"
51 INPUT F3
52 IF F3="" GOTO A0
53 LET Q8=1 "FLAG FILE"
54 FILE #2: F3
55 SCRATCH #2
56 PRINT "PLOT"
57 INPUT F3
58 IF F3="" GOTO A0
59 LET Q9=1 "FLAG NO-PLOT"
60 DIM C(150)
61
100 LIBRARY "PLOTLIB" (T0)
110
120 LET E=1.26402
130 LET V1=.0264306
140 LET V2=.0829632
150 LET V3=V1+V2
160 LET V4=V2+V3
170 LET V6=2*V2
180
200 DATA 0.0,0
205 DATA 1.-.2906776,-.2406776
210 DATA 0.0,0
215 DATA 1.-.4813552,0. 41
220 FOR I=1 TO 4
230 READ X(I),Y(I),Z(I)
235 IF Q8<>1 GOTO 240
240 WRITE #2: X(I),Y(I),Z(I)
245 NEXT I
250
260 READ N " # OF INTERVALS # OF PTS = 11"
265 IF Q8<>1 GOTO 270
265 WRITE #2:N
270 LET S=1/32 "STEP IN SCAN"
280 LET T=1.E-10 "TOLERANCE IN ZENOS"
290
300 * NOW DECIDE ON PLOT PARAMETERS, IF NEEDED
310 IF Q9=1 GOTO 600 "NOT NEEDED"
320 FOR I=1 TO 4
330 LET X0=X0+X(I)/4
340 LET Y0=Y0+Y(I)/4
350 LET Z0=Z0+Z(I)/4
360 NEXT I "X0,Y0,Z0 IS 'CENTER' OF FIGURE"
370 LET R9=0
380 FOR I=1 TO 4
390 LET R=(X(I)-X0)^2+(Y(I)-Y0)^2+(Z(I)-Z0)^2
400 IF R<R9 GOTO 420
410 LET R9=R "FIND MAX. DIAGONAL IN PLOT FOR SCALING"
420 NEXT I
430 LET R=SQRT(R9)
435
440 LET X1=(Y(3)-Y(2))*(Z(4)-Z(1)) - (Z(3)-Z(2))*(Y(4)-Y(1))
450 LET Y1=(Z(3)-Z(2))*(X(4)-X(1)) - (X(3)-X(2))*(Z(4)-Z(1))
460 LET Z1=(X(3)-X(2))*(Y(4)-Y(1)) - (Y(3)-Y(2))*(X(4)-X(1))

```

```

470 * M1,Y1,Z1 IS NORMAL VECTOR TO 'SURFACE' OF PTS. 1,2,3,4
480 LET A=10/SQR(X1^2+Y1^2+Z1^2) "SCALE FACTOR"
490 CALL "3SPHERE"(C1), X0,Y0,Z0, R0
495 PRINT " 3SPHERE RADIUS R0=";R0
500 CALL "3EYECART"(C1), X1-X1+A, Y0-Y0+A, Z0-Z1+A
510 DATA 1,2, 3,4, 1,3, 2,4
520 FOR I=1 TO 4
530 READ I1,I2
540 CALL "3CONNECT"(C1), X(I1),Y(I1),Z(I1), X(I2),Y(I2),Z(I2)
550 NEXT I
560
570
600 *NOW BEGIN CALCULATION
605
610 FOR E2=0 TO M "DO M+1 SCAN LINES"
615 LET U2=E2/M
617 LET U4=1-U2 "LINEAR PROPORTIONS"
620 LET X5=X(1)+U4 * X(3)+U2
622 LET Y5=Y(1)+U4 * Y(3)+U2
624 LET Z5=Z(1)+U4 * Z(3)+U2 "COORDS. OF INITIAL PT. ON LINE #E2"
626
628 LET X6=X(2)+U4 * X(4)+U2
630 LET Y6=Y(2)+U4 * Y(4)+U2
632 LET Z6=Z(2)+U4 * Z(4)+U2 "FINAL PT. ON LINE #E2"
634
636 LET R1=Y5+Y6+Z5
638 LET R2=X6+Y6+Z6-R1
640 LET F4=FNZ(R1,R2)
650 FOR E1=0 TO 1 STEP 5
660 LET F5=0
670 LET D=-1
680 LET A=E1
690 LET B=E1+5
700 LET F1=F4
710 LET F2=F4+FNZ(M,E2) "SAVE F4 FOR NEXT F1"
720 IF SGN(F1)=SGN(F2) GOTO 730
730 LET C=A+F1*(B-A)/(F1-F2)
740 LET F1=FNZ(C,E2)
750 IF ABS(F1)<T9 GOTO 800 "IF IN TOLERANCE"
760 IF F3=F5 GOTO 800 "IF WITHIN TRUNCATION ERROR"
770 LET F5=F3 "STORE OLD VALUE"
780 IF C=0 GOTO 800 "IF WITHIN TRUNCATION ERROR"
790 LET D=C
800 IF SGN(F1)=SGN(F1) GOTO 840
810 LET B=C
820 LET F2=F3
830 GOTO 730
840 LET A=C
850 LET F1=F3
860 GOTO 730
870
880 IF Q8=0 GOTO 910 "IF NOT DATA FILE"
885 LET R=(X1+X2+X3-R1)/R2 "TRY TO AVOID ZERO-ANGLE OVERFLOWS"
890 WRITE #2:R
900 IF Q9=1 GOTO 920 "IF NOT PLOTTING"
910 CALL "3DOT"(C1), M1,K2,M3 "COORDS. GENERATED BY LAST"
911 " CALL TO FNZ()".
920

```



```

922 GOTO 940 *ALLOW OUTPUT OF ONLY FIRST POINT FOUND
930 NEXT E1
940 NEXT E2
950 IF 09*1 GOTO 970
960 CALL "FINISH"(C1)
970 STOP
980
990
1000 DEF FNZ(E1,E2)
1010 LET U1=E1 * THE E1 COORD. IS ALREADY NORMALIZED TO 10,11
1020 LET K1=K5*U1 + M6+E1
1025 LET K2=Y5*U1 + Y6+E1
1030 LET K3=Z5*U1 + Z6+E1 *CART. COORDS. OF POINT
1040 LET FNZ=FN0(K1,K2,K3)
1050 FEND
1400 DEF FND(K1,K2,K3)
1410 LET K=K1+K1+K2+K2+K3+K3
1420 LET T0=K-E
1430 LET T1=K-2*(K1+K2+K3)+3-E
1440 LET T2=K-2*(K1+K2+K3)+3-E
1450 LET T3=K-4*K1-E
1460 LET F=(T0+T3-V1)*(T1+T2-V4)
1470 LET FND=F-V3*(T0+T3-V6)+T1+T2-V4
1480 FEND
1490
1500 END

```

EDATA-V1

```

10* LAST CHANGED 7.AUGUST 1974
12
14 * EDATA-V1 IS THE MAIN PROGRAM THAT CALCULATES ALL THE
16 * ELECTRON PROPERTIES FOR FERM1 SURFACE POINTS GENERATED
18 * BY FSEC-VY2. ALTERS TO THE PRESENT PROGRAM ALLOW IT
20 * TO PROCESS DATA FOR ANY REGION CORRECTLY. THIS VERSION
22 * WORKS FOR THE REGION V1 ON ZONE 2. ALL THE ALTER +1LS
24 * PIECES HAVE BEEN CONDENSED INTO ONE FILE TO SAVE
26 * TAPE STORAGE. SEE "ED-ALTS", WHICH FOLLOWS.
28
30 * THE FOLLOWING VARIABLES ARE "OVERLAP" WITH THEMSELVES, I.E.
32 * HAVE DIFFERENT USES IN DIFFERENT PLACES:
34 * U1,U2,U3,U4 M1,M2,M3, X1,X2, Y1,Y2, Z1,Z2
40
42 LIBRARY=PHYSLIB:="EIGEN"
100 DIM K(12,42),Y(12,42),Z(12,42),U(12,4),V(12,4),W(12,4)
101 DIM N(12),R(12,42),A(42),E(42),M(42,4)
102 DIM S(4,4),F(4,4),I(4),G(3,3),O(4,4)
110 LET H=41 *MAX DOWN INDEX ( # POINTS - 1)
120 LET H=11 *MAX ACROSS INDEX ( # FILES - 1)
140
155 LET E=1.26402
160 LET V1=0.026402
165 LET V2=0.0829837
170 LET V3=V1+V1
175 LET V4=V2+V2
190
200 DATA 1,1,1, 1,1,-1, 2,0,0
210 MAT READ G
220 LET S(1,2)=S(1,3)*S(2,4)*S(3,4)*V1
230 LET S(1,4)=S(2,3)*V2 *THESE ELEMENTS OF SECULAR MATRIX S(4,4)
240 * ARE CONSTANTS.
250
260 FOR I=2 TO 4
270 FOR J=1 TO 1-1
280 LET S(1,J)=S(J,1) *SYMMETRIZE IT
290 NEXT J
300 NEXT I
310
320 DATA V1-,RV1-
330 READ R1,R2 *INPUT, OUTPUT FILENAME ROOTS
340
350 * MAX *** NOW ALL INITIAL CONSTANTS ARE DEFINED
360
370
380 * READ IN ALL DATA
410 FOR I=1 TO N
420 FILE #1:R1&STAT(I)
430 FOR J=1 TO 4
440 READ M1(U(I,J),V(I,J),W(I,J) *SCAFFOLD COORDS.
450 NEXT J
460 READ M2:W(1) *# OF DATA PTS - 1
470 LET U1=U(1,1)
480 LET U5=V(1,1)
490 LET M1=W(1,1) *AVOID SUBSCRIPTS ON COMMONLY USED VARS.
510 LET U2=U(1,2)
520 LET U6=V(1,2)
530 LET M2=W(1,2) *DITTO

```

```

540 LET U3=U(1,3)-U1
550 LET U7=U(1,7)-U5
560 LET U3=U(1,3)-U1
570 LET U4=U(1,4)-U2
580 LET U8=U(1,4)-U6
590 LET U4=U(1,4)-U2
600 FOR J=0 TO N(1)
610 IF N(1)>0 GOTO 610
620 LET J1=0
630 IF ONLY 1 PT. IN FILE
640 GOTO 620
650 DIV/0 MESSAGE
660 LET J1=N(1)
670 READ #1:R(1),J1
680 LET R=R(1),J1
690 LET X1=U1+J1*U3
700 LET Y1=U5+J1*U7
710 LET Z1=N1+J1*W3
720 LET X2=U2+J1*U4
730 LET Y2=U6+J1*U8
740 LET Z2=N2+J1*W4
750 LET X(1,J)=X1 + R*X2
760 LET Y(1,J)=Y1 + R*Y2
770 LET Z(1,J)=Z1 + R*Z2
780 NEXT J
790 NEXT I
800
810 NOW DATA READ-IN IS ALL DONE
820
830 LET F=0
840 LET I=0
850 LET A(1)=FNA(1,0,0,1,0,0,1)/4
860
870 FOR I=1 TO N(F)-1
880 LET A=FNA(0,1-1,1,1,0,1)
890 LET A(1)=A+FNA(1,1,0,1+1,0,1)/4
900
910 NEXT I
920 LET J=N(F)
930 LET A=FNA(F,1-1,1,1,1,1,0)/8
940
950 GOSUB 1760
960 THEN DO OUTPUT,
970
980 FOR F=1 TO M-1
990 LET I=0
1000 LET A=FNA(F,1,1,1,1,1,0)
1010 LET A(1)=A+FNA(F,1+1,1,1,1,1,0)/4
1020
1030 FOR J=1 TO N(F)-1
1040 LET A=FNA(F,1-1,1,1,1,1,0)
1050 LET A(1)=A+FNA(F,1,1,1,1,1,0)/4
1060
1070 NEXT J
1080 LET I=N(F)

```

```

1090 LET A=FNA(F,1-1,1,1,1,1,0)
1100 LET A(1)=A+FNA(F,1,1,1,1,1,0)/8
1110 LET A(1)=A+FNA(F,1,1,1,1,1,0)/4
1120
1130 GOSUB 1760
1140
1150 NEXT F
1160
1170 LET F=M
1180
1190 LET F=M
1200 LET I=0
1210 LET A(1)=FNA(F,1-1,1,1,1,1,0)/4
1220 FOR J=1 TO N(F)-1
1230 LET A=FNA(F,1,1,1,1,1,0)
1240 LET A(1)=A+FNA(F,1,1,1,1,1,0)/4
1250
1260 NEXT J
1270 LET I=N(F)
1280 LET A=FNA(F,1,1,1,1,1,0)/4
1290 LET A(1)=A+FNA(F,1,1,1,1,1,0)/8
1300
1310 GOSUB 1760
1320
1330 STOP
1340
1350
1360
1370
1380 DEF FNA(I1,J1,I2,J2,I3,J3)
1390 LET X0=X(I1,J1)
1400 LET Y0=Y(I1,J1)
1410 LET Z0=Z(I1,J1)
1420 LET X1=X(I1,J1)+X0
1430 LET Y1=Y(I1,J1)+Y0
1440 LET Z1=Z(I1,J1)+Z0
1450 LET X2=X(I2,J2)+X0
1460 LET Y2=Y(I2,J2)+Y0
1470 LET Z2=Z(I2,J2)+Z0
1480 LET A1=X1+Y2-Z1+Y2
1490 LET A2=X2+Y2-X1+Z2
1500 LET A3=X1+Y2-Y1+X2
1510 LET FNA=SQR( FNA(A1,A2,A3) )
1520 FNA=0
1530
1540 DEF FNA(U,V,W)=(U+V+W)/3
1550
1560
1570 LET FNA(1,0,0)
1580 LET K1=Z(I1,J1)
1590 LET K2=Y(I1,J1)
1600 LET K3=X(I1,J1)
1610 LET X=FNA(K1,K2,K3)
1620 LET Y=X-E
1630 LET Z=X-2*(K1+K2+K3)+3+E
1640 LET T2=X-2*(K1+K2+K3)+3-E
1650 LET T3=X-4*K1+4-E
1660 LET U1=T0+T3-Y4

```

```

1670 LET U2=T1+T2-V*
1680 LET U3=T0+T1+T2+T3-4*V2
1690 LET U4=U1+(T1+T2)*U2-(T0+T3)-2*V3+U3
1700 LET W1=(X1+U4) * U1+(T1+T2)-2*U2+T0+2*V3+U3
1710 LET W2=(K2+U4) * U1+(T1+T2)+2*V3-(T0+T3-2*V2)
1720 LET W3=(K3+U4) * U1+(T1+T2)
1730 LET FME=2*SQR1 FNM(W1,W2,W3) 1/ABS(U4)
1740 FME=0
1750
1760 ' SUBROUTINE TO DO ROUTINE DATA FILE CALCULATIONS AND
1770 ' HANDLE OUTPUT
1780
1790 FOR I=0 TO NIF
1800 LET E(I)=FMEIF(I)
1810 GOSUB 1870 'DO MIXING COEFFS.
1820 NEXT I
1830
1840 FILE #3:RDS&STR$(F)$.M " OUTPUT FILENAME
1842 SCRATCH #3
1844 FOR Q0=2 TO 4 STEP 2 'SKIP GAMMA (0:0.0) COORDS -- FOR V1-
1846 ' FILES, ANYWAY (MAY CHANGE THIS)
1848 WRITE #3:U(F,Q0),V(F,Q0),M(F,Q0)
1849 NEXT Q0
1850 WRITE #3:N(F)
1852 FOR I=0 TO NIF
1854 WRITE #3:R(F,I),A(F,I),E(I),M(1,1),M(1,2),M(1,3),M(1,4)
1856 NEXT I
1860 LET Q0=LOC(#3)+1
1862 FILE #3:" " 'CLOSE IT AT ONCE
1864 PRINT "FILE #=";F; " LENGTH #";Q0
1865 PRINT " AT "TIM;" SEC."
1866 'ALL DONE WITH OUTPUT
1868 RETURN
1870
1875 '-----
1880
1890 ' SUBROUTINE: GIVEN I,J, CALCULATE MIXING COEFFS.
1900
1910 LET Q1=X(F+I)
1920 LET Q2=Y(F+I)
1930 LET Q3=Z(F+I)
1940 LET S(1,1)=FNM(Q1,Q2,Q3)-E
1950 FOR Q0=1 TO 3
1960 LET S(Q0+1,Q0+1)=FNM(Q1-Q(Q0,1),Q2-Q(Q0,2),Q3-Q(Q0,3)) * E
1970 NEXT Q0
1980 CALL "EIGEN":S(1,1),4,1(1,F),0
1990 ' NOW FIND LOWEST MAGNITUDE E-VALUE
2000 LET Q4=1 'ABSURD INITIAL GUESS AT MINIMUM E-VAL.
2010 FOR Q0=1 TO 4
2020 IF ABS(1(Q0)) > .04 GOTO 2050
2030 LET N4=Q0 'INDEX OF BEST CHOICE
2040 LET Q4=ABS(1(Q0))
2050 NEXT Q0
2060
2070 IF Q4 < 1.E-3 GOTO 2190 'IF WITHIN TOLERANCE
2080 PRINT "++FILE #";F; " DATUM #";I
2090 PRINT " EIGENVALUE #";Q4
2100 PRINT " EIGENVALUES ARE"

```

```

2110 MAY PRINT V1
2120 PRINT "X Y Z = "I;Q2;Q3
2130 PRINT "MATRIX PRODUCT IS"
2140 MAY Q=S*F
2150 MAY PRINT Q;
2160 PRINT
2170 PRINT
2180
2190 FOR Q0=1 TO 4
2200 LET M(1,Q0)=F(Q0,N4) 'COPY BEST E-VECTOR
2210 NEXT Q0
2220
2230 RETURN
2240
2250
2260 ' -----
2270
2280 END

```

ED-ALTS

```

100 8 * EDATA-V
110 10* THIS FILE IS A LIST OF ALTERS TO "EDATA-V" WHICH ALLOWS THE
120 12* PROGRAM TO PROCESS X-REGION DATA CORRECTLY. APART FROM
130 14* DIFFERENT FILENAME PARAMETERS; THE ONLY CHANGE IS IN THE
140 16* WAY THE AREA ELEMENT FOR THE FINAL POINT (AT X) IS
150 18* CALCULATED.
160 20
170 110 LET M=6
180 120 LET N=6
190 320 DATA X*, RX-
200 321* INPUT, OUTPUT FILENAME ROOTS
210 1210 LET A(I)=FNA(F-1,I, F-1,I+1, F,I)/8 *SINGLE TERM TERM AT CORNER
220 1220
230 1230
240 1240 * *** Wipe out these lines in EDATA-V1 LOOK JUST FOR
250 1250 * APPEARANCE'S SAKE
260 1260
270 1270
280 1280
290
300
310 2* EDATA-KL
320 1* THIS FILE IS A LIST OF ALTERS TO "EDATA-V" WHICH ALLOWS THE
330 2* PROGRAM TO PROCESS KL-REGION DATA. AREA ELEMENTS ARE CALCU-
340 3* LATED DIFFERENTLY BECAUSE OF THE SQUARE RATHER THAN TRIANGULAR
350 4* DATA ARRAY.
360 5
370 100 DIM X(19,19),Y(19,19),Z(19,19),U(19,4),V(19,4),N(19,4)
380 101 DIM N(19),R(19,19),A(19),E(19),M(19,4)
390 102 DIM M(19) *MOST OF THESE DIM'S JUST HAD TO BE INCREASED.
400 110 LET M=19
410 120 LET N=19
420 320 DATA KL-,RKI-
430 475 READ A(I,K(I)) *INTERPOLATION PARAMETER
440 476 LET A(I,K(I))=N(I)/M(I)+A(I,K(I))
450 477 LET B(I,K(I))=N(I)/M(I)+A(I,K(I))
460 478 ! 47. BY * INTERP. PARAMS.
470 602
480 603 ! DON'T NEED THESE LINES FROM EDATA-V1
490 624
500 610 LET J1=J+1+J+R7)
510 690 LET A(I)=FNA(F-1, F-1,I, F,I)/4 * 1 TERM AT CORNER
520 1090 LET A(FNA(F-1, F-1,I, F,I))
530 1100 LET A(I)=A(FNA(F-1, F-1,I, F,I))/4
540 1110 * 2 REGULAR TERMS AT BOTTOM OF INTERIOR LINES
550 1270 LET A(I)=FNA(F-1, F-1,I, F,I)/4
560 1280 * 1 REGULAR TERM AT "BOTTOM" CORNER
570 1871 WRITE #3:M(I)
580
590
600 10 * EDATA-SV
610 20* ZONE 2 V1-RIDGE ALTERS. USES KL-REGION CONNECTIVITY
620 22* BUT LINEAR INTERPOLATION, AND HEADER IS J2- INSTEAD OF
630 24* 6-WORD.
640 26
650 110 LET M=18
660 120 LET N=7 *CHANGED FROM 6 TO 7 ON 26.9.74
670 320 DATA V1-,SV1-

```

```

680 690 LET A(I)=FNA(F-1, F-1,I, F,I)/4 * ONE TERM AT CORNER
690 1090 LET A(FNA(F-1, F-1,I, F,I))
700 1100 LET A(I)=A(FNA(F-1, F-1,I, F,I))/4
710 1110 * 2 REG. TERMS AT BOTTOM OF INTERIOR LINES
720 1270 LET A(I)=FNA(F-1, F-1,I, F,I)/4
730 1280 * 1 REG. TERM AT "BOTTOM" CORNER
740 1844 FOR Q=1 TO 4 *WRITE ALL 4 SCAFFOLD POINTS
750
760
800 10 * EDAT-SV2
810 20 * EDI MER EDATA-V1:EDATA-SV:EDAT-SV2
820 22
830 110 LET M=18
840 120 LET N=6
850 320 DATA V2-,SV2-
860
870
880
900 10 * EDATA-SM
910 20* EDI MER EDATA-V1:EDATA-SV:EDATA-SM
920 22* ONLY FILE PARAMETERS ARE DIFFERENT HERE FROM EDATA-SV.
930 24
940 110 LET M=12
950 120 LET N=6
960 320 DATA W-,SW-
970
980 *** END OF JOINEU-TOGETHER-LIST-OF-FILES

```

```

      EIGEN
2  * SUBPROGRAM EIGEN  A D MEADOR, WINTER 1974
3
4  * CALLING SEQUENCE: CALL EIGEN(A(1),N,V(1),E1,1,0
5
6  * WHERE  A(1) = ORIGINAL MATRIX (LEFT UNCHANGED) WHOSE EIGEN-
7            VALUES/VECTORS ARE TO BE FOUND
8  *      + * NOTE: A(1) MUST BE SYMMETRIC, I.E. A(I,J)=A(J,I)
9            N    = DIMENSION OF MATRIX
10           V(1) = EIGENVALUES SORTED FROM HIGHEST VALUE TO LOWEST
11
12           E1,1 = EIGENVECTORS STORED COLUMNWISE (IN LEFT-TO-RIG
13
14           ORDER MATCHING EIGENVALUES V(1) )
15           D    = ACTIVITY COEF. 0 OR 1
16           D=0 -- CALCULATE EIGENVALUES AND EIGENVECTORS
17           D=1 -- CALCULATE EIGENVALUES ONLY
18
19
20 * DESCRIPTION : THIS BASIC SUBPROGRAM IS BASED ON THE FORTRAN
21
22 * SUBROUTINE "EIGEN" IN THE IBM SCIENTIFIC SUBROUTINES PACKA
23
24 * ONLY THE UPPER-TRIANGULAR HALF OF A(1) IS USED, THE OR
25 ORIGINAL
26 * MATRIX A(1) IS LEFT UNCHANGED, ONLY V(1) AND E1,1 ARE ALTE
27 ANED
28 * BY A CALL TO "EIGEN".
29 * CURRENTLY "EIGEN" CAN ACCEPT MATRICES OF DIMENSION N<=2
30
31
32 * FOR DIMENSION N=4, TYPICAL COMPUTATION TIMES ARE
33
34 * .09 SEC WITH D=0
35 * .06 SEC WITH D=1
36 * OF COURSE THIS TIME INCREASES RAPIDLY WITH INCREASING N.
37
38
39 * * * * *
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

100 LET R=1.0 E+4  *RANGE
101 IF D=1 GOTO 430
102 LET I1=0
103 FOR J=1 TO N  *GENERATE IDENTITY MATRIX
104 LET I1(I1)+N
105 FOR I=1 TO N
106 LET I2(I)=1
107 IF I<>J GOTO 390
108 LET O(I2)=1.  *DIAGONAL ELEMENTS
109 NEXT J
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

320 LET HMM=(M+1)
330 FOR I=1 TO M
340 LET I1=(I+1)-1/2
350 IF I=1 GOTO 380
360 IF I=M GOTO 380
370 IF I=1 GOTO 380
380 LET M9=M-I
390 GOTO 310
400 LET M7=M+I
410 IF I=M GOTO 340
420 LET L3=I+L1
430 GOTO 350
440 LET L3=L+I
450 LET B=Z(L3)*C5 - Z(M3)*S5
460 LET Z(M3)=Z(L3)+S5+Z(M3)*C5
470 LET Z(L3)=B
480 IF M=1 GOTO 1040
490 LET L9=L+I
500 LET M9=M+I
510 LET B=Q(L9)*C5-Q(M9)*S5
520 LET Q(M9)=Q(L9)+S5+Q(M9)*C5
530 LET Q(L9)=B
540 NEXT I
550 LET B=Z(L2)*S7
560 LET M=Z(L3)+C6+Z(M3)+S6 - 0
570 LET B=Z(L3)+S6+Z(M3)+C6 + 0
580 LET Z(L2)=Z(L3)+Z(M3)*S7 + Z(L2)*C6+36
590 LET Z(L3)=M
600 LET Z(M3)=0
610
620
630
640
650
660
670
680
690
700
710
720
730
740
750
760
770
780
790
800
810
820
830
840
850
860
870
880
890
900
910
920
930
940
950
960
970
980
990
1000
1010
1020
1030
1040
1050
1060
1070
1080
1090
1100
1110
1120
1130
1140
1150
1160
1170
1180
1190
1200
1210
1220
1230
1240
1250
1260
1270
1280
1290
1300
1310
1320
1330
1340
1350

```

```

1360 LET J1=M1+N
1370 LET M3=J1+J1/2
1380 IF Z(L3)+Z(M3) GOTO 1510
1390 LET B=Z(L3)
1400 LET Z(L3)=Z(M3)
1410 LET Z(M3)=B
1420 IF 0=1 GOTO 1510
1430 FOR K=1 TO M
1440 LET L9=J1+K
1450 LET M9=J1+K
1460 LET B=Q(L9)
1470 LET Q(L9)=Q(M9)
1480 LET Q(M9)=B
1490 NEXT K
1500
1510 NEXT J
1520 NEXT I
1530
1540
1550
1560
1570
1580
1590
1600
1610
1620
1630
1640
1650
1660
1670
1680
1690
1700
1710
1720
1730
1740
1750
1760
1770
1780
1790
1800
1810
1820
1830
1840
1850
1860
1870
1880
1890
1900
1910
1920
1930
1940
1950
1960
1970
1980
1990
2000
2010
2020
2030
2040
2050
2060
2070
2080
2090
2100
2110
2120
2130
2140
2150
2160
2170
2180
2190
2200
2210
2220
2230
2240
2250
2260
2270
2280
2290
2300
2310
2320
2330
2340
2350
2360
2370
2380
2390
2400
2410
2420
2430
2440
2450
2460
2470
2480
2490
2500
2510
2520
2530
2540
2550
2560
2570
2580
2590
2600
2610
2620
2630
2640
2650
2660
2670
2680
2690
2700
2710
2720
2730
2740
2750
2760
2770
2780
2790
2800
2810
2820
2830
2840
2850
2860
2870
2880
2890
2900
2910
2920
2930
2940
2950
2960
2970
2980
2990
3000
3010
3020
3030
3040
3050
3060
3070
3080
3090
3100
3110
3120
3130
3140
3150
3160
3170
3180
3190
3200
3210
3220
3230
3240
3250
3260
3270
3280
3290
3300
3310
3320
3330
3340
3350
3360
3370
3380
3390
3400
3410
3420
3430
3440
3450
3460
3470
3480
3490
3500
3510
3520
3530
3540
3550
3560
3570
3580
3590
3600
3610
3620
3630
3640
3650
3660
3670
3680
3690
3700
3710
3720
3730
3740
3750
3760
3770
3780
3790
3800
3810
3820
3830
3840
3850
3860
3870
3880
3890
3900
3910
3920
3930
3940
3950
3960
3970
3980
3990
4000
4010
4020
4030
4040
4050
4060
4070
4080
4090
4100
4110
4120
4130
4140
4150
4160
4170
4180
4190
4200
4210
4220
4230
4240
4250
4260
4270
4280
4290
4300
4310
4320
4330
4340
4350
4360
4370
4380
4390
4400
4410
4420
4430
4440
4450
4460
4470
4480
4490
4500
4510
4520
4530
4540
4550
4560
4570
4580
4590
4600
4610
4620
4630
4640
4650
4660
4670
4680
4690
4700
4710
4720
4730
4740
4750
4760
4770
4780
4790
4800
4810
4820
4830
4840
4850
4860
4870
4880
4890
4900
4910
4920
4930
4940
4950
4960
4970
4980
4990
5000
5010
5020
5030
5040
5050
5060
5070
5080
5090
5100
5110
5120
5130
5140
5150
5160
5170
5180
5190
5200
5210
5220
5230
5240
5250
5260
5270
5280
5290
5300
5310
5320
5330
5340
5350
5360
5370
5380
5390
5400
5410
5420
5430
5440
5450
5460
5470
5480
5490
5500
5510
5520
5530
5540
5550
5560
5570
5580
5590
5600
5610
5620
5630
5640
5650
5660
5670
5680
5690
5700
5710
5720
5730
5740
5750
5760
5770
5780
5790
5800
5810
5820
5830
5840
5850
5860
5870
5880
5890
5900
5910
5920
5930
5940
5950
5960
5970
5980
5990
6000
6010
6020
6030
6040
6050
6060
6070
6080
6090
6100
6110
6120
6130
6140
6150
6160
6170
6180
6190
6200
6210
6220
6230
6240
6250
6260
6270
6280
6290
6300
6310
6320
6330
6340
6350
6360
6370
6380
6390
6400
6410
6420
6430
6440
6450
6460
6470
6480
6490
6500
6510
6520
6530
6540
6550
6560
6570
6580
6590
6600
6610
6620
6630
6640
6650
6660
6670
6680
6690
6700
6710
6720
6730
6740
6750
6760
6770
6780
6790
6800
6810
6820
6830
6840
6850
6860
6870
6880
6890
6900
6910
6920
6930
6940
6950
6960
6970
6980
6990
7000
7010
7020
7030
7040
7050
7060
7070
7080
7090
7100
7110
7120
7130
7140
7150
7160
7170
7180
7190
7200
7210
7220
7230
7240
7250
7260
7270
7280
7290
7300
7310
7320
7330
7340
7350
7360
7370
7380
7390
7400
7410
7420
7430
7440
7450
7460
7470
7480
7490
7500
7510
7520
7530
7540
7550
7560
7570
7580
7590
7600
7610
7620
7630
7640
7650
7660
7670
7680
7690
7700
7710
7720
7730
7740
7750
7760
7770
7780
7790
7800
7810
7820
7830
7840
7850
7860
7870
7880
7890
7900
7910
7920
7930
7940
7950
7960
7970
7980
7990
8000
8010
8020
8030
8040
8050
8060
8070
8080
8090
8100
8110
8120
8130
8140
8150
8160
8170
8180
8190
8200
8210
8220
8230
8240
8250
8260
8270
8280
8290
8300
8310
8320
8330
8340
8350
8360
8370
8380
8390
8400
8410
8420
8430
8440
8450
8460
8470
8480
8490
8500
8510
8520
8530
8540
8550
8560
8570
8580
8590
8600
8610
8620
8630
8640
8650
8660
8670
8680
8690
8700
8710
8720
8730
8740
8750
8760
8770
8780
8790
8800
8810
8820
8830
8840
8850
8860
8870
8880
8890
8900
8910
8920
8930
8940
8950
8960
8970
8980
8990
9000
9010
9020
9030
9040
9050
9060
9070
9080
9090
9100
9110
9120
9130
9140
9150
9160
9170
9180
9190
9200
9210
9220
9230
9240
9250
9260
9270
9280
9290
9300
9310
9320
9330
9340
9350
9360
9370
9380
9390
9400
9410
9420
9430
9440
9450
9460
9470
9480
9490
9500
9510
9520
9530
9540
9550
9560
9570
9580
9590
9600
9610
9620
9630
9640
9650
9660
9670
9680
9690
9700
9710
9720
9730
9740
9750
9760
9770
9780
9790
9800
9810
9820
9830
9840
9850
9860
9870
9880
9890
9900
9910
9920
9930
9940
9950
9960
9970
9980
9990
10000

```

X-4OPW

```

0* D11001X-4OPW A B HEADR BACKUP COPY -- DLN 11/09/74
1* OPW-NEWV CONTAINS ALTERS TO X-4OPW FOR NEW (10.10.74) VIOF
2* VERSION OF 6. SEP. 1974 USING CORRECTED MATRIX ELEMENT STUFF
3* LINE 570 CHANGED ON 5.11.74 TO GET ALL UNITS ABSOLUTELY RIGHT
4*
50 PRINT "COMPILATION AT "TIM
60 PRINT "CLOCK TIME "CLKS
70 PRINT
80 MARGIN 120 *CALT
90 LIBRARY "PHYSLIB" *EIGEN" = F20115: X4OPWLIB9
100 DIM E(3,3), V(3), F(2), F(2), R(8,3), T(6,3)
110 DIM C(70), A(4), M(4), P(3), Q(3), K(8), Z(4)
120 DIM D(3,3), P(3), J(3,3)
130 DIM D(456)
140 DIM M(1)
150 DIM M(1)
160 DIM M(1)
170 DIM M(14) *HEADER DATA FOR A FILE
180 DIM L(48), M(49) *LAMBDA AREA CONTRIBS. *CALT
190
200 LET F3=7 *# OF FILE REGIONS PROGRAMMED
210 DATA 6,6,7,12,12,12,12
220 MAT READ N *# OF HEADER WDS. PER FILE
230 LET M(1)=1 *INITIALIZE IN CASE DEFAULT (LINEAR INTERP.)
240 CALL X4OPW-CONV(G(1),C(1),R(1),T(1),P(1),E,V,F(1),F(1),P(1),P(1),C(1),C(1),S(1))
245 LET S0=S(0,0) *GET NONSUBSCRIPTED COPY OF SYMMETRY (NO:K)
246 LET S1=S(0,1)
247 LET S2=S(0,2) *FILE REGION LIMITS ( L. 650 )
250
260
270 * GET INITIAL WAVEVECTOR P(1) AND MATR. ELEM. A(1) ----
280 FILE #1:=D11523:"F3(P(1))&STR$(P(2)) *P(1)'S FROM 4OPW-CON
290 FOR I=1 TO N(P(1))
300 READ #1:M(1) *READ HEADER DATA
310 NEXT I
320 READ #1:M2 *# OF DATA IN FILE -1
330 READ #1:1+M(P(1)+7-P(3)) *GET THE DESIRED DATUM
340 READ #1:24
350 READ #1:22,22,A(1),A(2),A(3),A(4) *SKIP "A", "V". READ MIXING COEFFS.
360 LET M1=F(1+P(1)) *COPY & SAVE REGION INDEX #
370 GOSUB 2120 *PROCESS HEADER DATA (F3 MUST BE DEFINED RIGHT)
380 LET I1=P(3)+S1 P(3)+M(1)+M(1)+M(1) *QUADR. INTERP. INDEX
390 GOSUB 2590 *USE K(1)-FORMAT RTZ ROUTINE FOR INIT. PT. P(1)
400 LET P(1)=M(1) *...AND COPY OVER.
410 LET P(2)=M(2)
420 LET P(3)=M(3)
421
422 PRINT " INITIAL POINT: P(1)=X(4)K(15)K(6)
423 PRINT " INTERPOLATED BETWEEN "P(1)M(1)M(2)M(3)" AND "I1
424 PRINT M(1)+M(4):M(2)+M(5):M(3)+M(6)
425 PRINT " WITH INTERPOLATION FRACTION 11 * F(1)
426
430
440
450 * DEFINE V(0) PARAMETERS P5,P6,P7
460 LET P4=.04/1 2+P1+P1+.52917+SORIE) *LAMBDA SQUARED
470 LET P5=1/(2+SORIE) *#-UNITS FOR V(1) CALC.
480 LET P5=P5+P5 *USE FOR A.V. SQUARED
490 LET P6=.611946

```

```

499
500 LET P4=P1+SORIE+P4/4.04 *P4
510 LET P7=P4+1 *PBETA=2
520 LET P8=(2/3)+E *ENERGY UNITS FOR V
530 LET P8=P4+P8 *MAKE IT A SINGLE PREFACTOR FOR CALCULATION.
540
550 LET T0=8000 *L O F S OF RUNTIME!!
560
570 LET C4=(6.02252*10^23*(E-5))/(2-36.9015)
571 * MASS OF ELECTRON/(2 MASS OF ALUMINUM ATOM)
580
590
600
610
620
630 * NOW BEGIN READING DATA FILES AND CALCULATING INTEGRAND
640
650 FOR F1=F5 TO F6 *LOOP ON FILE SERIES (ATCHES ON F.9.)
660 * COMPLETE CALCULATION IS F101 TO 7
670
680 MAT L=ZER(40)
690 MAT M=ZER(40) *CLEAR DEBUGGING COUNTERS
700
710 FOR F2=0 TO F(1) *FILE SER. & LOOP
720 LET F3:=D11523:"F3(F1)&STR$(F2)
730 FILE #1:=F3
740 PRINT F3,"TIM
750 FOR I=1 TO M(F1)
760 READ #1:M(1) *READ HEADER DATA
770 NEXT I
780 IF F3(8) GOTO 790 *IF AN INTERPOLATION PARAM. IS READ
790 LET M(13)=1 *ELSE SET TO DEFAULT VALUE
800 READ #1:M2 *# OF PTS.-1 IN DATA FILE
810 FOR I=0 TO M2 *NOW READ IN ALL DATA IN FILE
820 LET I1=I+7
830 FOR J=1 TO 7
840 READ #1:D(21+J)
850 NEXT J
860 NEXT I
870 FILE #1:=F3 *CLOSE DATA FILE
880 GOSUB 2120 *PROCESS HEADER DATA
890
900 FOR I=0 TO M2 *NOW MAIN LOOP ON DATA PTS.
910
920 GOSUB 2470 *PROCESS K V Z DATA
930
940 LET S1=S0+0 *SUB-SUM ON 48 EDU. POINTS
950 * NOW GENERATE THE 48 EQUIVALENT POINTS
960
970 FOR I1=1 TO 8 *REFLECTION OPERATIONS
980 FOR I2=1 TO 6 *PERMUTATIONS
990
1000 IF S0=0 GOTO 1010
1010 IF S(11,12)=0 GOTO 1080 *NEXT I2 IF NOT ENABLED
1020
1030 GOSUB 2010 *GET RUNTIME ELAPSED

```



```

2890 RETURN
2100
2110 * SUBROUTINE * - - - - - PROCESS HEADER DATA FROM A FILE
2120 IF F1<F3 GOTO 2180
2130 PRINT "HEADER SUBROUTINE FAILURE,"
2140 STOP
2150
2160 IF N(F1)=12 GOTO 2730 '12-ND. HEADER
2170 FOR Z1=4 TO 6
2180 LET M(Z1)=M(Z1)-M(Z1-3)
2190 NEXT Z1
2200
2210 GOTO 2280
2220
2230 FOR Z1=7 TO 12
2240 LET M(Z1)=M(Z1)-M(Z1-6)
2250 NEXT Z1
2260
2270 IF N2>0 GOTO 2120 'N OF PTS ALONG SCAFFOLD LINE
2280 LET M(13)=M(14)+0 'INTERP. WILL GIVE INIT. PT.
2290 RETURN
2300
2310
2320 IF F1=3 GOTO 2170
2330
2340 LET Z1=1 'LINEAR INTERP.
2350 GOTO 2390
2360
2370 LET Z1=M(7) 'Z1 IS LAST HEADER ND. FOR F1=3
2380
2390 LET M(13)=(1-Z1)/(M2-M2*(Z1+1)) 'QUADR. TERM
2400 LET M(14)=2*Z1/(M2*(Z1+1)) 'LINEAR TERM IN INTERPOLATION
2410 RETURN
2420
2430
2440
2450
2460 * SUBROUTINE * - - - - - GET X,Y,Z DATA FROM
2470 LET I=7 '1-DATUM INDEX FROM MAIN PROG.
2480 IF N2=0 GOTO 2510 'AVOID DIV/0 MESSAGE
2490 LET I1=1+M(13)+I + M(14) 'GENERAL QUADRATIC INTERPOLATION
2500 GOTO 2530 'SKIP N2=0 PATCH
2510 LET I1=0
2520
2530 IF F1<F3 GOTO 2570
2540 PRINT "XYZ SUBROUTINE FAILURE,"
2550 STOP
2560
2570 LET Z4=D(I+1) 'R-VALUE
2580
2590 'NOW DEFINE X-T-Z COORDS OF FINAL PT.1
2600 'NOTE SUBROUTINE ENTERED AT 2590 BY CALL FROM L.390
2610 IF M(F1)=12 GOTO 2690 '4-PT. INTERP. REQUIRED
2620
2630 LET K(4)=Z4*( M(2) + I1*M(4) )
2640 LET K(5)=Z4*( M(2) + I1*M(5) )
2650 LET K(6)=Z4*( M(3) + I1*M(6) )
2660 GOTO 2730

```

```

2670
2680
2690 LET K(4)=M(1) + Z4*( M(4)-M(1) ) + I1*( M(7) + Z4*( M(10)-M(7) ) )
2700 LET K(5)=M(2) + Z4*( M(5)-M(2) ) + I1*( M(8) + Z4*( M(11)-M(8) ) )
2710 LET K(6)=M(3) + Z4*( M(6)-M(3) ) + I1*( M(9) + Z4*( M(12)-M(9) ) )
2720
2730 LET A=D(I)+21
2740 LET Y=D(I)+3
2750 FOR Z5=1 TO 4
2760 LET B(Z5)=D(I)+3-Z51
2770 NEXT Z5
2780
2790 RETURN
2800
2810 LET T=TIM
2820 RETURN
2830
2840
2850 PRINT "HALTING AT DATUM #111 WITH SYMMETRY ELEMENT # "100(I1-1)+12
2860 PRINT "RUN-TIME "ATIM
2870 PRINT "CLOCK TIME "JCLKS
2880 PRINT "ACCUMULATED INTEGRALS:"
2890 PRINT "50 "JS
2900 PRINT " S1 "S1=AY 'MULT. BY DA. 1/DEL-E
2910 PRINT " S2 "S2=AY
2920 STOP
2930
2940
2950 END

```

X4OPWLIB

```

8 * X4OPWLIB LAST CHANGE 6.FEB 1975 TO ALLOW SYMMETRY = 2
10 * X4OPWLIB CHANGE 5.11.74 UNITS IN LINE 555 CORRECTED FINALLY
11 * LAST VERSION (29.8.74) CALLED 40PV-LIB AND THEN X4OPWLIB.
12
14 * B-SHELL FORCE CONSTANT SOLUTION IMPLEMENTED. TO RETRIEVE
16 * THE OLD 21-SHELL STUFF, SEE DOCUMENTED ALTERS UNDER "40PA-204"
18 * (INITIALIZATION OF THE FORCE CONSTANT DATA) AND REPLACE THE
20 * B-SHELL "ELCM" WITH THE 21-SHELL VERSION STORED ON TAPE.
22
100 SUB "40PA-COMM:R(1),C(1),P(1),F(1),E,V1,V2,S1,F1),P1,P2,C1,C2,S1,)
110 * DEFINE VARIOUS CONSTANTS
120
130
140 LET E=1.26402 *FERMI ENERGY
150 LET V1=.0264308 *V-111
160 LET V2=.0829832 *V-200
170 DATA 1.1,1.1,1.1,1.1,2.0,0
180 MAT READ G *LATTICE VECTORS
190
200
210 FILE #1:="D115211FCNST2" *24 FORCE CONSTS FOR B-SHELL MODEL
215 * *** CHANGE TO "FCNST1" IF 78-FORCE CONSTS TO BE USED
220 MAT INPUT #1:C(26)
225 * *** MUST SAY C(78) INSTEAD IF USING FCNST1 FILE.
230 FILE #1:=""
235
240
250 FILE #1:="D115211FILESPEC" *CONTAINS FILE SPECIFICATIONS
255 MAT INPUT #1:F(5)
260 FILE #1:=""
270
280 DATA 1.1,1.1, 1.1,1.1, 1.1,1.1, 1.1,1.1
290 DATA 1.1,1.1, 1.1,1.1, 1.1,1.1, 1.1,1.1
300 DATA 1.2,3.1, 1.3,2.1, 2.3,1.1, 3.2,1.1, 3.1,2.1, 2.1,3.1
310 MAT READ R,T *REFLECTION AND PERMUTATION MATRICES
320
330
340 LET P1=3.14159265
350 LET P2=P1/180
360
370 * DEFINE INITIAL WAVEVECTOR
380 PRINT "INITIAL WAVEVECTOR:"
390 PRINT "REGION #1: FILE SER.#, DATUM #1 SYM. INDEX1 FILE REGION#"
400 INPUT P(1),P(2),P(3),S8,FS,F0 *USE P(1) FOR TEMP. STORAGE
410
420 * USE FILE MANIPULATION CODE IN MAIN PROGRAM TO CONVERT
430 * P(1) TO COORDINATE DATA.
440
450
460
470 LET M=26.9815 *ALUMINUM ATOMIC MASS
480 LET H1=1.66643
490 LET C2=39.4784176*H1*M/2 *P1=P1 *UNITS OF FREQ/RAD/SEC *10+12
500 LET Z1=4.04 E-8 *AL. LATTICE PARAM.
510 LET Z2=2*(Y.1991 E+78)/11.054 6-27)
520 LET Z1=Z1/(2*P1)
530 LET Z2=Z2/(2*P1) *CONV.FACTOR,1/(RAD/SEC X 10+12) TO (CALC. UNITS)
540
550
560 LET C2=Z2/Z1

```

```

560 LET B1=2*( C(4)+2*C(9)+C(25)+2*C(26) )
570
580
590 DATA 1.3,4.1, 4.5,2.1, 6.0,0.1, 10.16,4.1, 17.24,0.1, 25.26,2
600 * TO USE 78 FORCE CONSTANT ELEM; ACTIVATE THE FOLLOWING LINES:
605 * DATA 27.2914
610 * DATA 30.33,0.1, 34.37,4.1, 38.41,0.1, 42.51,0.1, 52.55,4.1, 55.51,0
620 * DATA 62.04,4.1, 65.08,0.1, 69.72,4.1, 73.76,0.1, 77.78,2
630
640 FOR I=1 TO 6
642 * MAKE THAT 1 TO 16 IF IT'S THE 78-FORCE CONSTANT STUFF
650 READ Z7,Z8,Z6
660 FOR I=7 TO 28
670 LET C(25)=Z8+C(25) *MULTIPLY ALL FORCE CONSTANTS BY SAMP_6 FACTORS
680 NEXT I
690 NEXT I
695
700 * NOW SET UP SYMMETRY MATRIX S(8,8) WHICH SPECIFIES WHICH
710 * SYMMETRY OPERATIONS ARE ALLOWABLE FOR THIS 158 SYMMETRY
720
730 LET X(1)=1
740 LET X(2)=.375
750 LET X(3)=.125 *TEST POINT= FOR ALL TRANSFORMATIONS
760 MAT S=CON(8,8) *FILL INITIALLY WITH 1'S
770 IF S=0 GOTO 1110 *DO NONE OF THIS STUFF
780 * SKIP ALL THIS IF NO SYMMETRY -- ALL 48 OPS ENABLED.
790 FOR I1=1 TO 8
800 FOR I2=1 TO 4
810 FOR J1=1 TO 3
820 LET U(I1)=R(I1),J1=X(T(12,J1)) *SYM.TRANSF.
830 NEXT J1
840 LET S7=S8
850 ON S7 GOTO 870, 920, 870, 970, 870, 920, 870
860
870 * TEST X>Y
880 LET S7=S7-1
890 IF U(1)>U(2) GOTO 1030 *THIS 11,12 OK
900 GOTO 1010 *ELSE GO DISABLE THIS ONE
910
920 * TEST Z>0
930 LET S7=S7-2
940 IF U(3)>0 GOTO 1030 *THIS ONE OK
950 GOTO 1010 *ELSE DISABLE
960
970 * TEST Y>Z
980 LET S7=S7-4
990 IF U(2)>U(3) GOTO 1030 *OK
1000 *ALL DISABLES MEET NEXT LINE
1010 LET S(1,12)=0
1020
1030 * BRANCHES CONVERGE HERE; MAY HAVE ONE OTHER TEST FOR SAME 11,12
1040 * IN CASE THE SYMMETRY IS 4-FOLD (S8=3,5,6)
1050 IF S7=0 GOTO 850 *GO DO ANOTHER
1060 * NOTE THAT THE ENABLED STATE IS AN "AND" RESULT, NOT "OR"
1070 NEXT I2
1080 NEXT I1
1090
1100 * THE MATRIX S(1) IS NOW COMPLETE

```

```

1105
1110 LET S(0,0)*S8 'STOP AWAY FOR TRIP BACK TO MAIN PROGRAM
1115 LET S(0,1)*F5
1116 LET S(0,2)*F6 'STOP THESE TOO
1120
1130 SUBEND '-----
1140
1150
1160
1170
1180 SUB "FREQY(Q(1),N(1),E(1),C(1),C1,C2,P1
1190 ' CALCULATE FREQUENCIES AND POLARIZATIONS FOR PHONON Q(1),
1200 ' THIS SUBPROGRAM IS ADAPTED FROM QAK-R100= GNU.
1210
1220 LET N1=3 'DIMENSION
1230 DIM Z(4)
1240 CALL "ELEM"(Q(1),N(1),P1,3,C1,C1)
1250 CALL "LIGEN"(E(1),J,Z(1),E(1),0
1260 FOR I=1 TO N1
1270 LET N(I)=Z(I)*C2 'KEEP FREQUENCIES SQUARED, IN CALC. UNITS
1280 NEXT I
1290 SUMEND '-----
1300
1310
1320 ' *** *** *** FOLLOWING IS FILE "ELEM-BS" MERGED INTO QPW-LIB: ***
1330 ' SUBPROGRAM "ELEM" FILENAME "ELEM-BS" 15.6.74
1340 ' THIS IS THE N-SHELL DYNAMICAL MATRIX DEFINITION ROUTINE.
1350 ' IT USES ONLY 25 FORCE CONSTANTS AND GIVES FREQUENCIES
1360 ' WITH AN ERROR OF AS MUCH AS 1% FOR TRANSVERSE PHONONS. WITH
1370 ' RESPECT TO THE 21-SHELL MODEL VALUES.
1380 ' TO USE THIS: THE PROPER FORCE CONSTANTS MUST BE USED TOO.
1390
1400 SUB "ELEM"(Q(1),A(1),P1,N1,C0,F1)
1410
1420 LET Z=P1*Q(1)
1430 LET S1=SIN(Z)
1440 LET C1=COS(Z)
1450 LET S2=2*S1*C1
1460 LET C2=1-2*S1*S1
1470 LET S3=S2+C1 + S1*C2
1480 LET C3=C2+C1 + S1*S2
1490 LET S4=2+S2+C2
1500 LET C4=1+2*S2+S2
1510 LET S5=S3+C2 + C3*S2
1520 LET C5=C3+C2 - S3*S2
1530 LET S6=2+S3+C3
1540 LET C6=1-2+S3+S3
1550
1560 LET Z=P1*Q(2)
1570 LET T1=SIN(Z)
1580 LET U1=COS(Z)
1590 LET T2=2*T1*U1
1600 LET U2=1-2*T1*T1
1610 LET T3=T2+U1 + T1*U2
1620 LET U3=U2+U1 - T1*T2
1630 LET T4=2*T2*U2
1640 LET U4=1-2*T2*T2
1650 LET T5=T3+U2 + U3*T2

```

```

660 LET D5=D3+D2 - T3*T2
670 LET T6=2*T3*U3
680 LET U6=1-2*T3*T3
690
700 LET Z=P1*Q(3)
710 LET U1=SIN(Z)
720 LET E1=COS(Z)
730 LET U2=2*U1*E1
740 LET E2=1-2*U1*U1
750 LET U3=U2+E1 + U1*E2
760 LET E3=E2+E1 - U1*U2
770 LET U4=2*U2*E2
780 LET E4=1-2*U2*U2
790 LET U5=U3+E2 + E3*U2
800 LET E5=E3+E2 - U3*U2
810 LET U6=2*U3*E3
820 LET E6=1-2*U3*U3
830
840 LET Z=C0+F(1)*(2-D5*(D1+E1))+F(2)*(1-D1*E1)
850 LET Z=Z-F(4)*C7-F(5)*(D2+E2)+F(6)*(1-C2*D1+E1)
860 LET Z=Z+F(7)*(2-C1*(D2+E1)+D1*E2)
870 LET Z=Z+F(10)*(2-C2*(D2+E2))+F(11)*(1-D2*E2)
880 LET Z=Z+F(13)*(2-C3*(D1+E1))-F(14)*(2-C1*(D3+E3))
890 LET Z=Z+F(15)*(2-D7*E1-E3*D1)
900 LET Z=Z+F(17)*(1-C2*D2+E2)
910 LET A0=F(19)*(2-C3*(D2+E1+D1*E2))
920 LET A0=A0+F(20)*(2-C2*(D1+E3+U3*E1))
930 LET A0=A0+F(21)*(2-C1*(D3+E2+E3))
940 LET A0=A0-F(26)*(D4+E4)-F(25)*C4
950
960 LET A(1,1)=Z+A0
970 LET Z1=F(3)*S1*T1 + F(9)*E1*(S2*T1+S1*T2)
980 LET Z1=Z1+F(4)*S1*T1*E2 + F(12)*S2*T2
990 LET Z1=Z1+F(16)*(S1*T1 + S1*T3)
1000 LET Z1=Z1+F(18)*S2*T2*E2
1010 LET Z1=Z1+F(24)*E1*(S3*T2+S2*T3)
1020 LET Z1=Z1+F(27)*E2*(S1*T3 + S3*T1)
1030 LET Z1=Z1+F(27)*E3*(S1*T2 + S2*T1)
1040
1050 LET A(1,2)=A(2,1)=Z1
1060
1070
1080 LET Z=C0+F(1)*(2-D1*(E1+C1))+F(2)*(1-E1*C1)
1090 LET Z=Z-F(4)*D7-F(5)*(E2+C2)+F(6)*(1-D2*E1+C1)
1100 LET Z=Z+F(7)*(2-D1*(E2+C1+E1*C2))
1110 LET Z=Z+F(10)*(2-D2*(E2+C2))+F(11)*(1-E2*C2)
1120 LET Z=Z+F(13)*(2-D3*(E1+C1))+F(14)*(2-D1*(E3+C3))
1130 LET Z=Z+F(15)*(2-E3*C1+C3*E1)
1140 LET Z=Z+F(17)*(1-D2*E2+C2)
1150 LET A0=F(19)*(2-D3*(E2+C1+E1*C2))
1160 LET A0=A0+F(20)*(2-D2*(E1+C3+E3*C1))
1170 LET A0=A0-F(21)*(2-D1*(E3+C2+E2*C3))
1180 LET A0=A0-F(26)*(E4+C4)-F(25)*D4
1190
1200 LET A(2,2)=Z+A0
1210
1220 LET Z1=F(3)*T1*U1 + F(9)*C1*(T2*U1+T1*U2)
1230 LET Z1=Z1+F(5)*T1*U1*C2 + F(12)*T2*U2

```

```

2240 LET Z1=Z1+F(16)*T3*U1 + T1*U3)
2250 LET Z1=Z1+F(18)*T2*U2+U2
2260 LET Z1=Z1+F(24)*C1*(T3+U2+T2+U3)
2270 LET Z1=Z1+F(23)*C2*(T1+U3 + T3*U1)
2280 LET Z1=Z1+F(22)*C3*(T1+U2 + T2*U1)
2290
2300 LET A(2,3)=A(3,2)+Z1
2310 LET Z=C0*(1)*(2-E1*(C1+D1))+F(2)*(1-C1+D1)
2320 LET Z=Z-F(4)*E2*(5)*(C2+D2)+F(6)*(1-E2*(C1+D1))
2330 LET Z=Z+F(7)*(2-E1*(C2+U1+C1+U2))
2340 LET Z=Z+F(10)*(2-E2*(C2+D2))+F(11)*(1-C2+D2)
2350 LET Z=Z+F(13)*(2-E1*(C1+D1))+F(14)*(2-E1*(C3+U3))
2360 LET Z=Z-F(15)*(2-C1+D1-D3+C1)
2370 LET Z=Z+F(17)*(1-E2*(C2+D2))
2380 LET A0=F(19)*(2-S1)*C2+D1+C1+D2)
2390 LET A0=A0+F(20)*(2-E2*(C1+D3+C3+D1))
2400 LET A0=A0+F(21)*(2-E1*(C3+D2+C2+D3))
2410 LET A0=A0+F(26)*(C4+D4)-F(25)*64
2420
2430 LET A(3,3)=Z+A0
2440 LET Z1=F(31)*U1+S1 + F(9)*D1*(U7+S1+U1+S2)
2450 LET Z1=Z1+F(8)*U1+S1+D2 + F(12)*U2+S2
2460 LET Z1=Z1+F(18)*(U3+S1 + U1+C1)
2470 LET Z1=Z1+F(18)*U2+S2+D2
2480 LET Z1=Z1+F(24)*D1*(U3+S2+U2+S3)
2490 LET Z1=Z1+F(23)*D2*(U1+S3 + U3+S1)
2500 LET Z1=Z1+F(22)*D3*(U1+S2 + U2+S1)
2510
2520 LET A(1,3)=A(3,1)+Z1
2530
2540 SUBEND

```

FILESPEC 14 MAR 75 01:12

RV1-,NRX-,RKL-,SV1-,NSW-,SX-,SV2-
 11, 10,19,7, 12,2,6
 READY

FCONST1 14 MAR 75 01:08

1.000	350	E+4	,-1.273	000	E+3	, 1.127	650	E+4	, 3.468	000	E+3
-2.180	000	E+2	,-7.772	300	E+2	,-1.458	333	E+2	,-2.088	333	E+2
-4.176	667	E+2	, 1.475	000	E+2	,-4.600	000	E+1	, 1.935	000	E+2
2.587	000	E+2	, 4.830	000	E+1	, 2.200	000	E+1	, 7.890	000	E+1
-1.503	000	E+2	,-1.513	333	E+2	, 7.957	143	E+1	, 2.814	286	E+1
-2.714	286	E+0	, 2.057	143	E+1	, 3.085	714	E+1	, 6.171	429	E+1
2.320	000	E+2	, 5.000	000	E+0	,-7.350	000	E+1	, 6.000	000	E+0
-7.950	000	E+1	,-1.353	333	E+2	,-2.833	333	E+0	,-8.833	333	E+0
-3.533	333	E+1	,-1.144	000	E+2	,-3.160	000	E+1	,-4.000	000	E+0
-5.520	000	E+1	, 4.181	818	E+1	, 1.636	364	E+1	, 3.054	545	E+1
4.581	818	E+1	, 8.100	000	E+1	, 2.100	000	E+1	, 2.000	000	E+1
4.000	000	E+1	,-3.023	077	E+1	,-1.569	231	E+1	, 9.230	769	E-1
-6.230	769	E+0	,-8.307	692	E+0	,-2.492	308	E+1	,-4.892	308	E+1
9.230	769	E-1	, 3.000	000	E+0	,-1.038	462	E+1	,-7.166	667	E+0
-3.666	667	E+0	,-3.166	667	E+0	,-3.333	333	E-1	,-8.333	333	E-1
-1.666	667	E+0	, 4.200	000	E+1	,-1.000	000	E+0	, 4.300	000	E+1
2.688	235	E+1	, 1.555	882	E+1	, 1.455	882	E+1	, 1.941	176	E+1
4.144	118	E+1	, 1.555	882	E+1	, 1.000	000	E+0	, 2.426	471	E+1
-1.355	556	E+1	,-1.888	889	E+0	,-7.777	778	E+0	,-1.555	556	E+1
-3.300	000	E+1	, 2.000	000	E+0						

FCONST2 14 MAR 75 01:11

10107,	-1337,	11444,	2452,	-529,	-625,	-182
-148,	-296,	271,	321,	-50,	461,	227
198,	88,	142,	-109,	-64,	-94,	-111
12,	18,	36,	-534,	-116		

V-ALTS

```

10 * V-ALTS 1 B HEADOR FEB 1975
20 * THIS FILE IS A LIST OF ALTERS TO CHANGE THE V(Q) FUNCTION
22 * USED IN THE PROGRAM X-4OPW,
24
30 * "OPW-NEWV" IS LINES 100-140; TYPE:
32 * ,EDI EXT 100-140,EDI DES TO RETRIEVE "OPW-NEWV"
34
36 * "OPW-OLDV" IS LINES 200-250; TYPE:
38 * ,EDI EXT 200-250,EDI DES TO RETRIEVE "OPW-OLDV".
40
42
100 1 * OPW-NEWV CONTAINS ALTERS TO X-4OPW FOR NEW (10.10.74) V(Q)
110 490 LET P6:=61515
120 495 LET P9:=25.744
130 496 LET P9:=P9-P5 'THIS ALLOWS USE OF P5-SCALED VALUE OF  $\Delta$  SQUARED
140 2080 LET Y9:=P9*(COS(P6*X9)/(W9*P4+W7))-EXP(-W9/P9)
150
160
170
180
190
200 1 * OPW-OLDV CONTAINS ALTERS TO RESTORE X-4OPW TO OLD V(Q) FORM
210 2 * AFTER IT HAS BEEN MERGED WITH OPW-NEWV.
220 490 LET P6:=611546
230 495
240 496
250 2080 LET Y9:=P9*(COS(P6*X9)/(W9*P4+W7))

```

T-4OPW

```

1 * OPW-NEWV CONTAINS ALTERS TO X-4OPW FOR NEW (10.10.74) V(Q)
2 * AFTER IT HAS BEEN MERGED WITH OPW-NEWV.
3 * VERSION OF 4. SEPT. 1974 USING CORRECTED MATRIX ELEMENT STUFF
4 * LINE 570 CHANGED ON 5.11.74 TO GET ALL UNITS ABSOLUTELY RIGHT
5
10 * T-ALTS 1 B HEADOR 7.11.74
11 * THIS PROGRAM SEGMENT IS TO BE MERGED WITH X-4OPW. IT WILL
12 * CAUSE CALCULATION OF INVERSE LIFETIMES AS WELL AS LAMBDA AT
13 * THE SPECIFIED INPUT POINT.
14 * THESE EDITS MAKE USE ONLY OF "U" VARIABLES TO AVOID
15 * OVERLAP WITH PREVIOUSLY USED VARIABLES.
16 * THE CODE FOR CALCULATING LAMBDA HAS BEEN SLIGHTLY ALTERED IN
17 * LINES 1560-1580 TO TAKE ADVANTAGE OF THE MATRIX ELEMENT DDT
18 * PRODUCT.
19
20 * VARIABLES HAVE THE FOLLOWING MEANINGS:
21 * U FREQUENCY IN SEC-1
22 * U0 CONV. FACTOR TO GET SEC-1 FROM CALC. FREQ. UNITS
23 * U4 H-BAR OVER K-BOLTZMANN
24 * U5 H-BAR*OMEGA/K*8
25 * U6 Y=EXP(U5/T)
26 * U8 STEP IN TEMPERATURE (SET TO 5 DEGREES AT FIRST)
27 * U9 TEMPERATURE IN DEG K.
28
45
50 LET U0:=11.054 F-27/(2*(10.1091 E-28))-1*(2*3.14159265/4.04 E-8)+2
55 LET U4:=1.054 E-27/(1.38054 E-16)
60 LET U8:=5 'TEMP INC.
65 DIM U(500)
67 LET U1:=5 'INIT. TEMP.
68 LET U9:=100 'FINAL TEMP.
70
70 PRINT "COMPILATION AT "TIM
91 PRINT "CLOCK TIME "CLCKS
92 PRINT
93 MARGIN (20) 'CALL
100 LIBRARY=PHYSLIB+"LEIGEN",F201191X40PWLIB
110 DIM G(3,3),Y(3),F(7),F(7),R(8,3),T(6,3)
120 DIM G(78),A(4),B(4),P(3),O(3),K(6),X(6)
130 DIM G(3,3),P(3),J(8,3)
140 DIM B(46)
150 DIM Y(3)
160 DIM H(7)
170 DIM H(14) 'HEADER DATA FOR A FILE
180 DIM L(40),B(45) 'LAMBDA, AREA CONTRIBS. 'CALL
190
200 LET F3:=7 'N OF FILE REGIONS PROGRAMMED
210 DATA 0.6,7,12,17,32,32
220 MAT READ N 'N OF HEADER WDS. PER FILE
230 LET H(1)=1 'INITIALIZE IN CASE DEFAULT (LINEAR INTERP.)
240 CALL X4OPW-CON(1,1,C(1),R(1),T(1),P(1),E,V1,2,F(1),F(1),P1,P2,C1,C2,S(1)
245 LET S0:=S(0,0) 'GET NONSUBSCRIPTED COPY OF SYMMETRY INDEX
246 LET F5:=S(0,1)
247 LET F6:=S(0,2) 'FILE REGION LIMITS (L, 650)
250
260
270 * GET INITIAL WAVEVECTOR P(1) AND MATR. ELEMNS A(1) ----
280 FILE #1:"C115231"6F4(P(1))&STR$(P(2)) 'P(1)'S FROM 4OPW-CON

```

```

290 FOR I=1 TO N(P(1))
300 READ #1:M(1)      'READ HEADER DATA
310 NEXT I
320 READ #1:N2        'P OF DATA IN FILE =1
330 RESET#1: 1=N(P(1)):2=P(3)  'GET THE DESIRED DATUM
340 READ #1:Z4
350 READ #1:Z2,Z2,A(1),A(2),A(3),A(4)  'SKIP "A", "V", READ MIXING COEFFS
360 LET N1=F1:P(1)  'COPY & SAVE REGION INDEX #
370 GOSUB 2120  'PROCESS HEADER DATA (F1 MUST BE DEFINED RIGHT)
380 LET I1=P(3):P(3)+N(13) = N(14)  'QUADRANT INTERP. INDEX
390 GOSUB 2500  'USE K()-FORMAT XYZ ROUTING FOR INT. PT. P(1)
400 LET P(1)=K(4)  '...AND COPY OVER,
410 LET P(2)=K(5)
420 LET P(3)=K(6)
421
422 PRINT "  INITIAL POINT: "K(4)IN(5)IX(4)
423 PRINT "INTERPOLATED BETWEEN "I H(1)H(2)H(3))" AND "I
424 PRINT H(1)+H(4):K(6):H(5)H(3)+H(8)
425 PRINT " WITH INTERPOLATION FRACTION I1 = V(1)
426
430
440
450 ' DEFINE V(0) PARAMETERS P5,P6,P7
460 LET P6=1.04/(2*P1+P1+.52917+SOR(E))  ' "LAMBDA" SQUARED
470 LET P5=1/(2+SOR(E))  'K-UNITS FOR Y(1) CALC.
480 LET P5=P5*P5  'USE FOR M.V. SQUARED
490 LET P6=.61515
495 LET P9=.25.744
496 LET P9=P9*P9  'THIS ALLOWS USE OF P5-SCALED VALUE OF Q SQUARED
500 LET P6=4*P1/SOR(E)+P6/4.04  ' "SM"
510 LET P7=P4+L  ' PRETA**2
520 LET P8=(2/3)*E  'ENERGY UNITS FOR V
530 LET R8=P4*P8  'MAKE IT A SINGLE PREFACTOR FOR CALCULATION.
540
550 LET T9=6000  'L O T 9 OF RUNTIME!
560
570 LET G6=(.02252*(9.1081E-5)/(2*26.9815)
571 ' MASS OF ELECTRON/(2 MASS OF ALUMINUM ATOM)
580
590
600
610
620
630 ' NOW BEGIN READING DATA FILES AND CALCULATING INTEGRAND
640
650 FOR P1=P5 TO P6  'LOOP ON FILE SERIES (PATCHES ON F.S.)
660 ' COMPLETE CALCULATION IS F1=1 TO 7
670
680
690 MAT L=ZER(48)
690 MAT M=ZER(48)  'CLEAN DEBUGGING COUNTERS
690
700 FOR F2=0 TO F(F1)  'FILE SER. # LOOP
710 LET F3="011323:"&F(F1)&STR3(F2)
720 FILE #1:F3
730 PRINT F3;" "TIM
740 FOR I=1 TO N(F1)
750 READ #1:M(1)  'READ HEADER DATA
760 NEXT I

```

```

770 IF F1=J GOTO 790  'IF AN INTERPOLATION PARAM. IS READ
780 LET M(13)=1  'ELSE SET TO DEFAULT VALUE
790 READ #1:N2  'P OF PTS.=1 IN DATA FILE
800 FOR I=0 TO N2  'NOW READ IN ALL DATA IN FILE
810 LET I1=I+7
820 FOR J=1 TO 7
830 READ #1:O(2)+J)
840 NEXT J
850 NEXT I
860 FILE #1:"*F  'CLOSE DATA FILE
870 GOSUB 2120  'PROCESS HEADER DATA
880
890
900 FOR I=0 TO N2  'NOW MAIN LOOP ON DATA PTS.
910
920 GOSUB 2470  'PROCESS K Y Z DATA
930
940 LET S1=S4+0  'SUB-SUM ON 48 EQU. POINTS
950 ' NOW GENERATE THE 48 EQUIVALENT POINTS
960
970 FOR I1=1 TO 8  'REFLECTION OPERATIONS
980 FOR I2=1 TO 8  'PERMUTATIONS
990
995 IF S8=0 GOTO 990
996 IF S(1,12)=0 GOTO 1000  'NEXT I2 IF NOT ENABLED
997
998 GOSUB 2210  'GET RUNTIME ELAPSED
1000 IF T>T9 GOTO 2050  'TIME'S UP.
1010 LET Z3=0  'D NORM
1020 FOR I2=1 TO 3  'DEFINE FINAL ELECTRON Q.V. AND UNREDUCED PHONON M.V.
1030 LET K(22)=R(11,22)+K1 T(12,22)+S)
1040 LET O(22)=P(22)-K(22)
1050 LET Z3=Z3+ABS(O(22))  'ACCU. NORM
1060 NEXT I2
1070 IF Z3<.0081 GOTO 1080  'NEXT SYM.ELEM. IF Q = APPROX. ZERO
1080
1090
1100 '*** SKIP A PLACE FOR THE REDUCED-Q CALC. ***
1110 CALL "FREQV40(1,K1),E(1),C(1),C1,C2,P1
1120 ' NOW QMGAS AND POLARIZATION VECTORS ARE DONE
1124
1125 IF ABS(M(1))>1.E-3 GOTO 1150  'NOT ZERO -- O.K. TO CONTINUE
1126 LET M9=M9+1  'COUNT PATHOLOGIES
1127 LET A9=A9+A  'ACCU. TOTAL AREA LOST
1128 GOTO 1680  'SKIP TO NEXT SYM. OP.
1129 ' *** THIS ASSUMES M(1)=0 <=> ALL 3 ZERO
1130
1140 ' GENERATE SYMMETRY-EQUIVALENT Q-VECTORS J(1)
1150 FOR J1=1 TO 3
1160 FOR J2=1 TO 3
1170 LET J(J1,J2)=R(11,J2)+G( J1, I(12,J2) )
1180 NEXT J2
1190 NEXT J1
1200 ' NOW CALCULATE
1210 LET M1=M2+M3+0  'CLEAR M.E. COMPONENTS FOR NEW BUNCH
1220
1230 ' NOW CALCULATE THE 16 TERMS OF THE M.E.
1240

```



```

2130 PRINT "HEADER SUBROUTINE FAILURE,"
2140 STOP
2150
2160 IF N1/I1=12 GOTO 2230 '12-ND. HEADER
2170 FOR I1=4 TO 6
2180 LET H(Z1)+H(Z1)-H(Z1-3)
2190 NEXT Z1
2200
2210 GOTO 2260
2220
2230 FOR I1=7 TO 12
2240 LET H(Z1)+H(Z1)-H(Z1-6)
2250 NEXT Z1
2260
2270
2280 IF N2>0 GOTO 2320 'N OF PTS ALONG SCAFFOLD LINE
2290 LET H(13)+H(14)+0 'INTERP. WILL GIVE INIT. PT.
2300 RETURN
2310
2320 IF F1=3 GOTO 2370
2330
2340 LET Z1=1 'LINEAR INTERP.
2350 GOTO 2390
2360
2370 LET Z1=H(1) 'IS LAST HEADER ND. FOR F1=3
2380
2390 LET H(13)+H(1-21)/(N2+H2*(Z1+1)) 'QUADR. TERM
2400 LET H(14)+2*Z1/(N2*(Z1+1)) 'LINEAR TERM IN INTERPOLATION
2410 RETURN
2420
2430
2440
2450
2460 ' SUBROUTINE ' - - - - - GET X,Y,Z DATA FORM
2470 LET I0=701 'I=DATUM INDEX FROM MAIN PROG.
2480 IF N2=0 GOTO 2510 'AVOID DIV/0 MESSAGE
2490 LET I1=I0+H(13)+1 + H(14) 'GENERAL QUADRATIC INTERPOLATION
2500 GOTO 2530 'SKIP N2=0 PATCH
2510 LET I1=0
2520
2530 IF F1=3 GOTO 2570
2540 PRINT "XYZ SUBROUTINE FAILURE,"
2550 STOP
2560
2570 LET Z1=0(I0+1) 'R-VALUE
2580
2590 ' NOW DEFINE X-Y-Z COORDS OF FINAL PT.
2600 ' NOTE SUBROUTINE ENTERED AT 2590 BY CALL FROM L.390
2610 IF N1/I1=12 GOTO 2690 '4-PT. INTERP. REQUIRED
2620
2630 LET K(4)=Z1+H(1) + I1*H(4)
2640 LET K(5)=Z1+H(1) + I1*H(5)
2650 LET K(6)=Z1+H(1) + I1*H(6)
2660 GOTO 2730
2670
2680
2690 LET K(4)+H(1) + Z1+H(4)+H(1) + I1*H(7) + Z1+H(10)-H(7)+I1
2700 LET K(5)+H(2) + Z1+H(5)+H(2) + I1*H(8) + Z1+H(11)-H(8)+I1

```

```

2710 LET K(6)+H(3) + Z1+H(6)+H(3) + I1*H(9) + Z1+H(12)-H(9)+I1
2720
2730 LET A=0(I0+2)
2740 LET V=0(I0+3)
2750 FOR Z5=1 TO 4
2760 LET B(25)=0(I0+3+Z5)
2770 NEXT Z5
2780
2790 RETURN
2800
2810 LET T=T+1
2820 RETURN
2830
2840
2850 PRINT "HALTING AT DATUM #",I1," WITH SYMMETRY ELEMENT # ",I0+(I1-1)*I2
2860 PRINT "RUN-TIME ",T+1
2870 PRINT "CLOCK TIME ",CLKS
2880 PRINT "ACCUMULATED INTEGRALS,"
2890 PRINT "S=","S
2900 PRINT " S1="S1+A/Y 'MULT. BY D1, 1/DEL-E
2910 PRINT " S2="S2+B/Y
2920 STOP
2930
2940
2950 END

```

T3

```

10 * T3 A B HEADOR OCT 1974
12 * CALCULATES THE T-CUBED COEFFICIENT OF RELAXATION TIME FOR
14 * THE 4-OPAL ALUMINUM MODEL. USES THE LIBRARY "K4OPWL18"
16 * WHICH IT ASSUMES IS IN THE CATALOG FOR F28119 (CURRENTLY).
18
20 * *** THIS PROGRAM IS STILL BEING DEBUGGED *** J0.1.75
22
50 DIM S(4,3),G(4,4),C(16,4),N(7),FS(7),F(7)
60 DIM M(3),E(3,3),X(78),Q(3),M(14)
70 DIM O(150),P(4,50)
75 DIM Q(32),V(17)
100 LPRINT "F28119:K4OPWL18", "PHYSLIB***:EISEN"
110 LPRINT "F28119:K4OPWL18***:TD1"
120
130 FILE #1:=D:\1523\FCONST2"
140 MAT INPUT #1:K(78)
150 FILE #1:=0"
160 LET #1:=J.14159265
170 LET #2:=J.4784376+1.66043+26.9815/(4+P1+P1)
180 * FREQUENCY IN HAT/SEC = 10+12
190 LET K3=(4.04 * #1)/(2+P1)
200 LET K1=2*(5.1091 E-28)/(1.054 E-27)
210 LET K3=K1*K3*(3-1E12) * CONVERTS OLD FREQ. UNITS TO CALC. UNITS
220 LET K2=K3/K2 * FINAL FORM OF FREQUENCY UNIT * ACTOR
230 LET K1=2*(K(4)+2*K(15)+K(25)+2*K(26))
240
250
260 DATA 1.3,4, 4.3,2, 6.9,8, 10.16,4, 17.24,8, 25.26,2
270 * TO USE /8 FORCE CONSTANT ELEM. ACTIVATE THE FOLLOWING LINES:
280 * DATA 27.29,4
290 * DATA 30.73,8, 34.37,4, 38.41,8, 42.51,8, 52.55,4, 55.51,8
300 * DATA 62.84,4, 65.68,8, 69.72,4, 73.76,8, 77.76,2
310
320 FOR Z=1 TO 6
330 * MAKE THAT 1 TO 18 IF IT'S THE 78-FORCE CONSTANT STUFF
340 READ Z,28,26
350 FOR Z=27 TO 28
360 LET K(25)=26*K(25) * MULTIPLY ALL FORCE CONSTANTS BY SIMPLE FACTORS
370 NEXT Z
380 NEXT Z
390
400 LET E=1.26402
410 LET N0=5
420 LET Y0=-1/3 * MINUS SIGN ADDED 24.2.75
430 * DEFINE SYMMETRIC VECTOR-MATRIX S(,)
440 DATA 9,0,0, 1,1,1, 1,1,-1, 2,0,0
450 MAT READ S * S IS ACTUALLY THE LATTICE VECTORS G
460 FOR I=1 TO 3
470 FOR J=1 TO 4
480 LET I1=J-1+(0-1)*(I-1)/2
490 LET S=0
500 FOR K=1 TO 3
510 LET G(I1,K1+S(J,K))-S(I,K)
520 LET S=S+G(I1,K)+G(I1,K)
530 NEXT K
540 LET G(I1,4)=SOR(S) * MAGNITUDE OF G(I1, 1-3)
550 NEXT J
560 NEXT I
570 NEXT I

```

```

580
590 DATA .0264386, .0264386, .0829832, .0829832, .0264386, .0264386
600 DATA .271002, .271002, .154669, .154669, .271002, .271002
620
630 MAT READ Y * Y(I, 1 TO 6) AND V(I,1, 1 TO 6)
640
650
660
670 PRINT "FILE REGION #, FILE #, DATUM #, PLOT FLAG"
680 INPUT F1,F2,F3, P0
690 LET B0=1.1888E-4 * UNITARY PREFACTOR FOR TOTAL CALCULATION
700 LET S0=S0+2*P1+7.185
710 IF P0*(P0-1)<0 GOTO 670
720 PRINT
730
740 DATA 6.6,7,12-12,12,12
750 MAT READ N * HEADER DATA
760 FILE #1:=D:\1523\FILESPEC"
770 MAT INPUT #1:FS * FILENAMES
780 MAT INPUT #1:F * FILES PER FILENAME
790 FILE #1:=0"
800
810 DATA 2,3,4, 1,0,5, 8,7,6 * NEIGHBOR INDICES IN FILE INPUT ORDER
820 MAT READ I(3,3)
830
840 FOR F5=1 TO 3
850 LET F6=F2-2+F5 * FILE #, IF DEFINED
860 FILE #1:=D:\1523\FILES(F1+8*F5)
870 IF (F5#1,"NUMERIC")<1 GOTO 1030 * SKIP IF NOT THERE
880 FOR I=1 TO K(F1)
890 READ #1:H(1) * GET THE HEADER DATA
900 NEXT I
910 READ #1:N2 * # OF DATA STORED
920 GOSUB 2020 * PROCESS HEADER DATA
930
940 FOR F6=1 TO 3
950 LET F9=F3-2+F6 * DATUM # IF DEFINED
960 LET I5=I(F5+F6) * NEIGHBOR INDEX
970 IF F9*(N2-F9)<0 GOTO 1030 * NO SUCH DATUM (C0 OR D2)
980 LET Y(I5)=1 * THIS NEIGHBOR EXISTS
990 RESET #1:N(F1)+1+70F9
1000 READ #1:Z4,Z5,U(1),C(15,1),C(15,2),C(15,3),C(15,4)
1010 LET I1=F9*(F9+M(1)) + M(14) * INTERPOLATION INDEX
1020 GOSUB 3150 * GET XYZ AT LOC. I5
1030 NEXT F6
1040
1050 NEXT F5
1060
1070 IF Y(I1)=0 GOTO 1100 * SKIP NEXT BLOCK
1080 LET T(9)=Y(I1)
1090 LET X(9)=X(I1)
1100 LET Y(9)=Y(I1)
1110 LET Z(9)=Z(I1)
1120 LET U(9)=U(I1)
1130 LET C(9,1)=C(I1,1)
1140 LET C(9,2)=C(I1,2)
1150 LET C(9,3)=C(I1,3)
1160 LET C(9,4)=C(I1,4)

```

```

1170
1180 FILE #1=" " 'NOW ALL DATA INPUT IS DONE.
1190
1200
1210 FOR I=1 TO 9
1220 LET X(I)=X(I)-X(0) ' TRANSLATE TO ORIGIN AT CENTER
1230 LET Y(I)=Y(I)-Y(0)
1240 LET Z(I)=Z(I)-Z(0)
1250 NEXT I
1260
1270
1280 LET U0=U(0) ' DEL-E AT ORIGIN
1290
1300
1310 FOR L1=1 TO 8 ' MAIN CALC. LOOP ON TRIANG. INTERVALS
1320 IF T(L1)+T(L1+1)≠0 GOTO 2210 ' NEXT INT. IF UNDEFINED ENOPOINT
1330 LET M1=SQRT(T(L1)*T(L1+1)+Y(L1)*Y(L1+1)+Z(L1)*Z(L1+1))
1340 FOR L2=0 TO 40 ' LOOP ON SUBINTERVAL WITHIN TRIANGLE
1350 LET L3=L2/40
1360 LET X=X(L1)+L3*(X(L1+1)-X(L1))
1370 LET Y=Y(L1)+L3*(Y(L1+1)-Y(L1)) ' COORDS. OF PHONON VECTOR
1380 LET Z=Z(L1)+L3*(Z(L1+1)-Z(L1))
1390 LET D1=X*X(L1)+Y*Y(L1)+Z*Z(L1) ' DOT PRODUCT
1400 LET M2=SQRT(M1*M1+D1) ' MAGN. OF D
1410 LET C1=D1/(M1*M2) ' COSθ ANGLE WITH INITIAL SIDE OF TRIANGLE
1420 IF C1<0 GOTO 1440 'TEST FOR NEGATIVE RADICAL
1430 LET D1=1-SONE-FIXES IT'S JUST A TEEINY BIT OVER
1440 LET B(L2)=ATN(SQRT(1-C1*C1)/C1) ' ANGLE θ L2
1450
1460 ' NOW CALCULATE INTEGRAND AT THIS POINT
1470 FOR I=1 TO 4 ' FIRST, GET LOCAL MIXING COEFFS
1480 LET M(I)=C4(L1)+L3*(C4(L1+1)-C4(L1)) ' SAME FORM AS FOR XYZ
1490 LET G(I)=(M(I)-C4(0))/M2 ' IDEALV. OF 1TH MIXING COEFF.
1500 NEXT I
1510
1520 ' NOW GET THE PHONON STUFF
1530 LET Q(1)=X
1540 LET Q(2)=Y
1550 LET Q(3)=Z
1560 CALL "RED"=Q(1):M(1):E(1):K(1):K1:K2:P1
1570 LET A(1)=A(2)=A(3)=0
1580 FOR J=1 TO 3
1590 FOR K=1 TO 4 ' NOW CALC. THE VECTOR M.C. OBJECT A(K)
1600 LET I1=J-1+(K-1)*4+(I-1)/2
1610 LET A1=C(I0+1)+Q(I1)*C(I0,J)+D(I1)
1620 ' NEXT 6 LINES, PART OF CONNECTION TERM OF 27.2.75
1630 LET Z3=0
1640 FOR K1=1 TO 3
1650 LET Z3=Z3+Q(K)*G(I1,K)
1660 NEXT K
1670 LET Z3=Z3/(G(I1,4)*M2) 'DIV.// MAGNITUDES OF Q & G
1680 LET Z3=Z3+Z4(C(I0,J)+Y(I1,4)) 'THE V-DERIVATIVE TERM
1690 LET Z2=V(I1)+C(I0,J)*M(J)+C(I0,J)*M(I1)/M2 'CORRECTION OF 25.2.75
1700 FOR K1=1 TO 3
1710 LET A(K)=A(K)+A1*G(I1,K)+V(I1)
1720 LET A(K)=A(K)-Z2*Q(K) 'THIS CORRECTION TERM ADDED ON 25.2.75
1730 LET A(K)=A(K)-Z3*G(I1,K) 'CORRECTION TERM ADDED 27.2.75
1740 NEXT K

```

```

1750 NEXT J
1760 NEXT I
1770
1780 FOR K=1 TO 3
1790 LET A(K)=A(K)-V0*Q(K)/M2 'ADD ON THE G-MAT TERM
1800 NEXT K
1810 ' NOW A(I) IS THE VECTOR TO BE DOTTED WITH EIGENVECTORS E(I)
1820 LET S3=S3+0 'CLEAR 4- & 1-OPH SUM COUNTERS
1830 FOR J=1 TO 3 ' LOOP ON POLARIZATION INDEX
1840 LET S4=S4+0
1850 FOR K1=1 TO 3
1860 LET S4=S4+E(K,J)*A(K) ' THE DOT PRODUCT
1870 LET S5=S5+E(K,J)*V3*Q(K)/M2 ' 1-OPH RESULT DONE TOO.
1880 NEXT K
1890 LET S4=S4-S4 ' 1-OPH
1900 LET C4=M(J)/(M2*M2) ' M(I) IS OMEGA SQUARED
1910 LET C4=C4+C4*U0 'PUT DEL-E ON TO C4 TERM
1920 LET S5=S5+S4/C4
1930 NEXT J
1940
1950 ' NOW S3 IS THE INTEGRAND FOR THIS PHONON Q
1960
1970 LET L(L2)=S3 ' STORE 4-OPH INTEGRANDS ALONG INTERVAL IN L(I)
1980 LET J(L2)=S5 ' STORE 1-OPH TOO
1990
2000 NEXT L2 ' NEXT INTERPOLATED POINT IN THIS TRIANGLE
2010
2020
2030 ' NOW ALL THE INTEGRANDS AND ANGLES FOR THIS REGION ARE TABULATED.
2040
2050 FOR L2=1 TO N0
2060 IF P0=0 GOTO 2120 'SKIP PLOT STORAGE
2070 LET I8=I8+1 'PLOT INDEX
2080 LET P(1,I8)=B(L2-1)+05
2090 LET P(2,I8)=L(L2-1) '4OPH VAL.
2100 LET P(3,I8)=J(L2-1) '1OPH VAL.
2110 LET P(4,I8)=L(L2-1)/J(L2-1) 'RATIO
2120 LET S2=S2+(B(L2)-B(L2-1))*L(L2)+L(L2-1)/2 ' 4-OPH SUM
2130 LET S7=S7+(B(L2)-B(L2-1))*J(L2)+J(L2-1)/2 ' 1-OPH SUM
2140 ' THIS IS SIMPLE TRAPEZOIDAL STUFF.
2150 NEXT L2
2160
2170 LET B5=B5+B(N0)
2180 ' NOW ALL DONE WITH THIS TRIANGULAR INTERVAL.
2190
2200
2210 NEXT L1
2220
2230 PRINT " 4OPH INTEGRAL "150=S2/B5;" 1OPH INTEGRAL "150=S7/B5
2240 PRINT "RATIO 4:1 = "152=S7/S2
2250 IF P0=1 GOTO 2280 'CONTINUE WITH PROGRAM
2260 STOP 'ELSE DON'T CONTINUE
2270
2280 LET I8=I8+1 'TACK ON LAST VAL.
2290 LET P(1,I8)=B5

```

```

2300 LET P(2,18)=L(2)
2310 LET P(3,18)=J(2)
2320 LET P(4,18)=L(2)/J(2)
2330 LET P8=18
2340 PRINT P8: "VALUES TO PLOT."
2350
2360
2370 * NOW DO PLOTTING
2380 PRINT
2390 PRINT " PLOT 40PW, 10PW, 411 RATION"
2400 INPUT Q(2),Q(3),Q(4)
2410 FOR Q=2 TO 4
2420 IF Q(4)=0 GOTO 2480
2430 * DR10=1 TO P8
2440 IF P(2,18) < R(2) GOTO 2460
2450 LET R(2)=P(2,18)
2460 NEXT 18
2470 LET R(2)=1.2*R(2) *SCALE TO LEAVE MARGIN
2480 NEXT 0
2490 IF R(2) < R(3) GOTO 2520 * FIND MAX(40PW,10PW)
2500 LET R(3)=R(2)
2510 GOTO 2530
2520 LET R(2)=R(3)
2530
2540
2550 FOR Q=2 TO 4
2560 * P(2,18)=0 GOTO 2790
2570 PRINT "READY FOR PLOT"
2580 LET P8
2590 CALL "LINES":Q(1), -R(2),R(2), -R(3),R(3)
2600 CALL "SQUARE":Q(1), 1
2610 LET Q(1)=LOG(R(2))/LOG(10)
2620 LET H2=INT(R(1))
2630 LET R1=R1+H2
2640 LET Q(1)=R1
2650 LET R1=.2*INT(H2*.999)/2)
2660 LET R1=R1+1 (10*H2)
2670 PRINT "SCALE UNIT =":R1
2680 IF ABS(1-1)+ABS(Q(2)-1) = 0 GOTO 2700 *DON'T PLOT SAME AXIS
2690 CALL "AXIS":Q(1), 0,R1,Q,R1
2700
2710 FOR 18=1 TO P8
2720 LET R=P(2,18)
2730 LET 1=P(1,18)
2740 CALL "LINE":Q(1), -R*SIN(1),-R*COS(1)
2750 NEXT 18
2760
2770 CALL "LIFT":Q(1)
2780 CALL "FINISH":Q(1)
2790 NEXT 0
2800
2810 STOP
2820 * SUBROUTINE - - PROCESS FILE HEADER DATA - - - - -
2830
2840 IF H(F1)=12 GOTO 2910 *12-ND, HEADER
2850 FOR Z1=4 TO 6
2860 LET M(Z1)=M(Z1)-M(Z1-3)
2870 NEXT Z1

```

```

2880
2890 GOTO 2960
2900
2910 FOR Z1=7 TO 12
2920 LET H(Z1)=M(Z1)-M(Z1-6)
2930 NEXT Z1
2940
2950
2960 IF M2>0 GOTO 3000 *# OF PTS ALONG SCAFFOLD LINE
2970 LET M(13)=M(14)=0 *INTERP, WILL GIVE INIT. PT.
2980 RETURN
2990
3000 IF F1=3 GOTO 3070
3010
3020 LET Z1=1 *LINEAR INTERP.
3030 GOTO 3070
3040
3050 LET Z1=M(7) *#7 IS LAST HEADER NO. FOR F1=3
3060
3070 LET H(13)=(1-Z1)/(M2*M2*(Z1+1)) *QUADR. TERM
3080 LET H(14)=2*Z1/(M2*(Z1+1)) *LINEAR TERM IN INTERPOLATION
3090 RETURN
3100
3110
3120
3130
3140
3150 * SUBROUTINE - - GET XYZ DATA FORM - - - - -
3160
3170 IF H(F1)=12 GOTO 3250 *4-PT. INTERP. REQUIRED
3180
3190 LET X(15)=Z4*( H(1) + 11*H(4) )
3200 LET Y(15)=Z4*( H(2) + 11*H(5) )
3210 LET Z(15)=Z4*( H(3) + 11*H(6) )
3220 GOTO 3240
3230
3240
3250 LET X(15)=H(1) + Z4*( H(4)-H(1) ) + 11*H(7) + Z4*( H(10)-H(7) )
3260 LET Y(15)=H(2) + Z4*( H(5)-H(2) ) + 11*H(8) + Z4*( H(11)-H(8) )
3270 LET Z(15)=H(3) + Z4*( H(6)-H(3) ) + 11*H(9) + Z4*( H(12)-H(9) )
3280
3290
3300 RETURN
3310
3320 END

```



```

1050 READ #11 24.75, U(15), C(15,1), C(15,2), C(15,3), C(15,4)
1060 LET (1+FF-1) * F9 * M(13) + M(14) ) ' INTERPOLATION INDEX
1070 GOSUB 2660 ' GET XYZ AT LOC, IS
1080 NEXT F6
1090
1100 NEXT F5
1110
1120 IF Y(1)=0 GOTO 1230 ' SKIP NEXT BLOCK
1130 LET T(9)=T(1)
1140 LET X(9)=X(1)
1150 LET Y(9)=Y(1)
1160 LET Z(9)=Z(1)
1170 LET U(9)=U(1)
1180 LET C(9,1)=C(1,1)
1190 LET C(9,2)=C(1,2)
1200 LET C(9,3)=C(1,3)
1210 LET C(9,4)=C(1,4)
1220
1230 :ILE #1:=" " ' NOW ALL DATA INPUT IS DONE.
1240
1250
1260 FOR I=1 TO 9
1270 LET X(I)=X(I)-X(0) ' TRANSLATE TO ORIGIN AT CENTER
1280 LET Y(I)=Y(I)-Y(0)
1290 LET Z(I)=Z(I)-Z(0)
1300 NEXT I
1310
1320
1330 LET U0=U(0) ' DEL-E AT ORIGIN
1340
1350
1360 FOR L1=1 TO 8 ' MAIN CALC. LOOP ON TRIANG. INTERVALS
1370 IF T(L1)-T(L1+1)=0 GOTO 2150 ' NEXT INT. IF UNDERNEED ENDOPOINT
1380 LET X1=SON( X(L1)+X(L1+1)-Y(L1)-Y(L1+1)-Z(L1)-Z(L1+1) )
1390 FOR L2=0 TO N0 ' LOOP ON SUBINTERVAL WITHIN TRIANGLE
1400 LET L3=L2/N0
1410 LET X=X(L1) + L3*( X(L1+1)-X(L1) ) /2
1420 LET Y=Y(L1) + L3*( Y(L1+1)-Y(L1) ) /2 ' PHONON W.V. AT BORDER
1430 LET Z=Z(L1) + L3*( Z(L1+1)-Z(L1) ) /2 ' OF 0TH CELL
1440 LET D1=X*X(L1+1)+Y*Y(L1+1)+Z*Z(L1+1) ' DOT PRODUCT
1450 LET M2=SON( X=X + Y*Y + Z*Z ) ' MAGN. OF Q
1460 LET C1=D1/(M1*M2) ' COS( ANGLE WITH INITIAL SIDE OF TRIANGLE
1470 IF C1<0 GOTO 1490 ' YES! FOR NEGATIVE RADICAL
1480 LET C1=1 ' SOMETIMES IT'S JUST A FEWNT BIT OVER
1490 LET B(L2)=ATN( SON(1-C1+C1)/C1 ) ' ANGLE @ L2
1500
1510 ' NOW CALCULATE MIXING COEFFICIENTS
1520 FOR I=1 TO 4
1530 LET M(I)=C(0,I) + C(L1,1) + L3*( C(L1+1,1)-C(L1,1) ) /2
1540 NEXT I ' TAKE AVERAGE OF CENTRAL C(I) AND INTERPOLATED BORDER C(I).
1550
1560 ' NOW THE PHONON QUANTITIES!
1570 LET Q(1)=X
1580 LET Q(2)=Y
1590 LET Q(3)=Z
1600 CALL "FRED"=Q(1),M(1),E(1), K(1),K1,K2,P1
1610 ' NOW CALCULATE THE VECTOR MATRIX ELEMENT THING
1620 LET D(1)=D(2)=D(3)=0

```

```

1630 FOR I=1 TO 4
1640 FOR J=1 TO 4
1650 FOR K=1 TO 3
1660 LET S(K)=6*(I2,K)+Q(1,K)+Q(K1
1670 NEXT K ' S(I) IS VECTOR K=K1-Q(1)+Q(12)
1680 GOSUB 2830
1690 FOR K=1 TO 3 ' COMPONENTS OF VECTOR FOR THIS POLARIZATION J
1700 LET D(K)=Q(K)+C(0,I)+M(12)+S(K)*V9
1710 NEXT K
1720 NEXT I2
1730 NEXT I1
1740 LET R2=R3=0
1750 MAT R=ZER(15) ' CLEAR SUMS
1760 FOR J=1 TO 3 ' LOOP ON POL. INDEX
1770 LET C0=M(I1)/M2*M2 ' SOUND VELOCITY SQUARED (CALC. UNITS)
1780 LET Y0=SON( M(J) ) * M2 ' H-BAR*OMEGA/K-B
1790
1800 FOR I7=14 TO 15 STEP 16
1810 CALL "EXP"=INT(1/Y0/I7, U(I7))
1820 NEXT I7
1830
1840 ' NOW CALCULATE THE CAPITAL-LAMBDA OBJECT
1850 LET L=D(1)*E(1,J) + D(2)*E(2,J) + D(3)*E(3,J)
1860 LET L=L/M(J) ' DOT-PRODUCT SQUARED OVER OMEGA SQUARED
1870 ' NOW LAMBDA-THING, L, IS CALCULATED
1880 LET LB=L/C0
1890 FOR I7=14 TO 15 STEP 16
1900 LET R(I7)=R(I7) + L*U(I7) ' POL. SUM FOR TEMPERATURE VECTOR
1910 NEXT I7
1920 LET R2=R2+L*M2 ' I -> INFINITY LIMIT
1930 LET R3=R3+L/C0 ' I -> 0 LIMIT
1940 NEXT J ' END OF POLARIZATION LOOP
1950
1960 ' NOW QUANTITIES AT THIS SUBINTERVAL ARE CALCULATED
1970 FOR I7=14 TO 15 STEP 16
1980 LET L(I7,L2)=R(I7) ' STORE ALL THE R(I)'S FOR EACH SUBINTERVAL
1990 NEXT I7
2000 LET J(L2)=R2
2010 LET V(L2)=R3 ' STORE THEM 'TIL END OF TRIANGULAR SECTION
2020 NEXT L2
2030
2040 FOR L2=1 TO N0 ' NOW SUM INTEGRAND*ANGLES
2050 LET B2=B(L2)-B(L7+1) ' DELTA THETA
2060 FOR I7=14 TO 15 STEP 16
2070 LET A(I7)=A(I7) + R2*( L(I7,L2)-L(I7,L2-1) ) /2
2080 NEXT I7 ' INTEGRATE THE TEMPERATURE VECTOR
2090 LET A2=A2 + B2*( J(L2)-J(L2-1) ) /2
2100 LET A3=A3 + B2*( V(L2)+V(L2-1) ) /2
2110 NEXT L2
2120
2130 LET B5=B5+B(N0) ' KEEP UP WITH TOTAL ANGLE
2140
2150 NEXT L1 ' NEXT TRIANGULAR ELEMENT
2160
2170 MAT A=( 2*P1/B5)*M1/U0 ) * A
2180 LET A2=(A2/B5)*2*P1 + M2/U0 ' GET RIGHT PHYSICAL UNITS
2190 LET A3=(A3/B5)*2*P1 + M3/U0
2200

```

```

2210
2220
2230 PRINT "LOW-T LIMIT/T+3 = "JAS
2240 PRINT
2250 PRINT * T 0TH ELEMENT 0TH ELEMENT/T+3 LARGE-T LIMIT*
2260 PRINT " L.T /T+3"
2270 FOR T7=T4 TO T5 STEP T6
2280 PRINT USING X1: (T, A(T7)+T7*T7*T7, A(T7)-A2*T7, A2/(T7*T7))
2290 IF F4=0 GOTO 2290 'SKIP FILE OUTPUT IF NO FILE
2300 PRINT #4:STR$(T7),",",STR$(A(T7))
2310 NEXT T7
2320 PRINT
2330
2340 STOP
2350 ' SUBROUTINE - - PROCESS FILE HEADER DATA - - - - -
2360 IF N(F1)=12 GOTO 2420 '12-NO. HEADER
2370 FOR Z1=4 TO 6
2380 LET H(Z1)=H(Z1)-H(Z1-3)
2390 NEXT Z1
2400 GOTO 2470
2410
2420 FOR Z1=7 TO 12
2430 LET H(Z1)=H(Z1)-H(Z1-6)
2440 NEXT Z1
2450
2460
2470 IF N2>0 GOTO 2510 *# OF PTS ALONG SCAFFOLD LINE
2480 LET H(13)=H(14)=0 'INTERP. WILL GIVE INIT. PT.
2490 RETURN
2500
2510 IF F1=3 GOTO 2560
2520
2530 LET Z1=1 'LINEAR INTERP.
2540 GOTO 2580
2550
2560 LET Z1=M(7) '07 IS LAST HEADER NO. FOR F1=3
2570
2580 LET H(13)=(1-Z1)/(N2-N2*(Z1+1)) 'QUADR. TERM
2590 LET H(14)=2*Z1/(N2*(Z1+1)) 'LINEAR TERM IN INTERPOLATION
2600 RETURN
2610
2620
2630
2640
2650
2660 ' SUBROUTINE - - GET XYZ DATA FORM - - - - -
2670
2680 IF N(F1)=12 GOTO 2760 '4-PT. INTERP. REQUIRED
2690
2700 LET X(15)=Z4+( H(1) + (1+H(4)) )
2710 LET Y(15)=Z4+( H(2) + (1+H(5)) )
2720 LET Z(15)=Z4+( H(3) + (1+H(6)) )
2730 GOTO 2800
2740
2750
2760 LET X(15)=H(1) + Z4+( H(4)-H(1) ) + (1+H(7) + Z4+( H(10)-H(7)))

```

```

2770 LET Y(15)=H(2) + Z4+( H(5)-H(2) ) + (1+H(8) + Z4+( H(11)-H(8)))
2780 LET Z(15)=H(3) + Z4+( H(6)-H(3) ) + (1+H(9) + Z4+( H(12)-H(9)))
2790
2800
2810 RETURN
2820
2830
2840
2850 ' SUBROUTINE CALCULATE Y(S(ZP),S(ZP+1),S(ZP+2))=Y9
2860 LET X9=S(1) 'THESE S()'S MUST BE DEFINED RIGHT BEFORE G3900.
2870 LET Y9=S(2)
2880 LET Z9=S(3)
2890 LET W9=( X9-X9+Y9+Y9+Z9+Z9 )=P5
2900 LET X9=SQR(W9) ' "X"
2910 LET W9=(W9/( P7 + 2*W9 ) ) * G(K)"
2920 LET M/=.5 + (1-W9)*LOG( ABS( (1-X9)/(1+X9) ))+W9/( 4*X9 ) 'F(X)"
2930 LET V9=P8+( COS( P4*X9 )/( W9+P4+W7 ) )=EXP( +W9/P4 )
2940 RETURN
2950
2960 END
2970 SUB "EXP-INT":Y0,F0
2980 IF Y0>2/3 GOTO 3020
2990 LET F0=(1/2)*LOG( 1+ Y0+Y0/6 )
3000 GOTO 3110
3010
3020 ' ELSE MUST CALCULATE A SERIES SOLN. TO INTEGRAL
3030 FOR N=1 TO 90
3040 LET H=2*M-1
3050 LET A=H*Y0
3060 LET S=(2*(1+H*M*M))/( 1 - 6*X9*(-A)+( 1+ A*( 1+A/2) ) )
3070 NEXT N
3080
3090 LET F0=S
3100
3110 SUBEND

```

Appendix B

Errors in the λ Calculation

1. Coarseness of the Data Grid

Since λ is calculated on the surface grid in the simplest way, just by adding up individual area elements multiplied by integrands, there is a significant possibility that the grid will be inadequate; the integrand may vary wildly between grid points. To check that this variation was not too severe, contour plots were prepared of sections of the λ integrand over the Fermi surface. After an initial data - grid (not otherwise referred to in this work) was actually found to be too coarse, the contour - plot technique was used to construct the final data grid. Examples of these contour plots are given in Figures B.1 - B.6. The criterion used in establishing the grid was that variation of the λ integrand from one point to its neighbors not exceed 10% of the integrand value.

It is interesting that this criterion becomes useless in the neighborhood of $q=0$. There, the integrand exhibits such drastic angular dependence that reducing the grid spacing only aggravates the problem. Thus, the region immediately around $q=0$ was not included in applying the 10% criterion. Instead, the area element around $q=0$ was omitted from the calculation altogether, since the integrand is not even well - defined at

Figure B.1. Contour plot of the λ integrand $\left[\frac{2m}{MN} \sum_{\sigma} \Lambda_{kk'}^{\sigma} \right]$ from (3.1) for the "KL" region of zone 2 (see Fig. 3.4) with the initial wavevector k at the point nearest K . The data-grid of Fig. 3.4 is displayed as a square array; the horizontal and vertical scales thus represent grid points rather than physical distance. The contour interval is 0.25, in the computational units discussed in Chapter III.

Figure B.2. Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector $\mathbf{k}$ in the $k_z=0$ plane, near K. The contour interval is 0.25.

See Fig. B.6 (b) for another plot with $\mathbf{k}$ in the $k_z=0$ plane.

Figure B.3. Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector k in the middle of this region. The contour interval is 0.05; hence the variation is not really as great as it appears in comparison with Figures B.1 and B.2.

Figure B.4. Contour plot of the λ integrand for the "KL" region of zone 2 with the initial wavevector k at the point nearest L . The contour interval is 0.1 .

Figure B.5. Contour plot of the integrand for the "X", "V1", and "V2" regions of zone 3 (see Fig. 3.6), with the initial wavevector k at the corner of the "V2" region nearest K (i.e. k at the point labelled #1 in Fig. 3.14). The contour interval is 0.5.

(a) 8 shells.

(b) 21 shells.

Figure B.6. Contour plots of the χ integrand for the "KL" region of zone 2 with the initial wavevector k at a point near W in the $k_z=0$ plane. In (a) the integrand is calculated using the 8-shell phonon model; in (b), using the 21-shell model. The contour interval is 0.25 in both cases.

$q=0$. Since the integrand does not become very large around $q=0$, such an omission caused only negligible error in the total integral.

With integrand variation less than 10% away from $q=0$, and indeed considerably less than 10% over most of the Fermi surface, it seems very safe to assign an accuracy of 10% to the integral calculation itself, apart from possible systematic errors in the physical models used.

2. The phonon model

As mentioned in Chapter III, two phonon models were available, one more accurate than the other. Final values of λ calculated from these two models agreed within 1.5% at the few points tested. The less accurate model was chosen therefore for the bulk of the calculations because it required significantly less computer time. Figure B.6 shows the variation in the integrand for λ around an initial point ("3,5,0") in the basal plane of zone 2 near W. The values of λ , calculated over just the Fermi - surface region shown, differ by less than 1%. From this example, it is clear that the phonon model is not a source of significant error.

It is worth noting here that the type of phonon model used is suspect near $q=0$. However, plots of ω versus q showed near - perfect linearity around $q=0$ for wavevectors far smaller than the data - grid spacing. For instance, in the [1 2 3]

direction, serious deviations from linearity occur only below $q \approx 0.0003$; this is roughly 3% of the grid spacing.

3. Choice of the form-factor $V(q)$

As discussed in Chapter III, there is latitude in the choice of the form factor. The values for λ reported in Chapter III used the Ashcroft form factor as described there. However, it was felt that better agreement ought to be obtained between $V(q)$ at the reciprocal lattice vectors and the values of V_{111} and V_{200} used to construct the Fermi surface. To get this agreement it was necessary to multiply the form factor (3.18) by a Gaussian:

$$V_{\text{new}}(q) = V_{\text{old}}(q) e^{-q^2/D} \quad (\text{B.1})$$

The parameter R_c in (3.18) was taken to be 0.6152 Å; the parameter D in (B.1) was taken to be 25.74. These choices gave $V(111) = 0.02650$ and $V(200) = 0.08284$, i.e. errors of respectively 0.4% and 1.7% with respect to the values given in (3.4). The two forms for $V(q)$, V_{old} and V_{new} , are plotted in Figure B.7 using the choices of parameters discussed here and in Chapter III.

It was hoped at first that this modification would lead to values of λ that differed by no more than a percent from the previous values. Instead, the new values differ by 20% from the old ones (see Table B.1). This discrepancy is at least

Initial wavevector location *	$\lambda_{old} V$	$\lambda_{new} V$	$\frac{\lambda_{old} - \lambda_{new}}{\lambda_{new}} \cdot 100\%$
3 19 0	0.5310	0.4408	20.5
3 19 1	0.5264	0.4372	20.4
3 19 5	0.4533	0.3801	19.3
3 19 10	0.3929	0.3328	18.1
3 19 19	0.3676	0.3132	17.4
7 0 0	0.3710	0.3158	17.5
7 4 0	0.3971	0.3347	18.6
4 4 0	0.3526	0.3014	17.0
6 2 0	0.3899	0.3324	17.3

*See Figs. 3.13 and 3.14 for an explanation of the "location" indices.

Table B.1. Values of λ calculated from the old form factor (3.18) and the new one (B.1), showing a rather uniform change of 20%.

rather constant over the Fermi surface, so that the results of Chapter III may be in error by a constant scale factor without altering the anisotropies exhibited. No comparisons were made for τ^{-1} calculations; these all employed the modified $V(q)$ of (B.1).

With the ambiguity of Chapter III results due to the energy - dependence of V_{111} and V_{200} , it is difficult to choose between the two $V(q)$ forms in fitting experimental m^*/m data. However, if the electron velocities v_k were corrected for that energy dependence, the sensitivity of λ to $V(q)$ might provide a very interesting test for a really correct choice of $V(q)$, which might then be plausibly extended to other calculations.

References

1. For a brief treatment of the Brillouin - Wigner theory, see G. Baym, Lectures on Quantum Mechanics (W. A. Benjamin, New York, 1973).
2. J. W. Wilkins, Observable Many-Body Effects in Metals (Nordita, Copenhagen, 1968).
3. J. R. Schrieffer, Theory of Superconductivity (W. A. Benjamin, New York, 1964).
4. See, for instance, P. L. Taylor, A Quantum Approach to the Solid State (Prentice-Hall, Englewood Cliffs, New Jersey, 1970), Chapter 5.
5. J. M. Ziman, Principles of the Theory of Solids (Cambridge University Press, Cambridge, 1964) presents a standard description of the O.P.W. formalism (pp. 93 - 97). See also Ashcroft, Reference 7 below.
6. See Ziman, op. cit., pp. 129-131.
7. N. W. Ashcroft, Phil. Mag. 8, 2055 (1963).
8. G. A. Korn and T. M. Korn, Mathematical Handbook for Scientists and Engineers (McGraw-Hill Book Company, New York, 1961), section 26.2.2.
9. S. V. F. Waite and D. G. Mather, eds., BASIC - Sixth Edition (University Press of New England, Hanover, N. H., 1972). The numeric random - access file structure is discussed in section 3.2.
10. N. W. Ashcroft, Phys. Letters 23, 48 (1966).
11. See, for instance, J. A. Reissland, The Physics of Phonons (John Wiley & Sons, New York, 1973), Chapt. 1.

12. W. M. Hartmann and T. O. Milbrodt, Phys. Rev. B₃, 4133 (1971).
13. L. J. Raubenheimer and G. Gilat, A FORTRAN Program for Calculating Frequency - Distribution Functions of Cubic Crystals, Oak Ridge National Laboratory (June 1966), ORNL-TM-1425.
14. Up-to-date lists of the force constants were provided by W. M. Hartmann (private communication, 1974).
15. N. W. Ashcroft, Ph. D. Thesis (Cambridge University, 1964). The table of m^*/m values follows page 54.
16. This lifetime applies in principle to a single electronic state k on the Fermi surface; experimentally it can be related to the linewidth of a cyclotron resonance spectrum. Thus, like λ , it is determined experimentally only over an orbit and not at individual points. (The resistivity lifetime, by contrast, relates to an excitation of all the states on the Fermi surface; it is proportional to T^5 instead of T^3).
17. In addition to the Wilkins book (ref. 2), see also D. Nowak and M. J. G. Lee, [Phys. Rev. B₅, 2851 (1972)] for a calculation in copper; and D. K. Wagner, R. C. Albers, and M. Ray [Cornell University - Materials Science Center Technical Report # 2172, March, 1974] for a calculation in potassium.