Molecular Dynamics Calculation of Mean Square Displacement in Alkali Metals and Rare Gas Solids and Comparison with Lattice Dynamics

by

Gernot A. Heiser, B. Sc., Freiburg (Germany)

A Thesis
submitted to the Department of Physics
in partial fulfilment of the requirements
for the degree of
Master of Science

June 1984
Brock University
St. Catharines, Ontario

to Kurt

Non est ad astra mollis e terris via

Seneca

ABSTRACT

Molecular dynamics calculations of the mean square displacement have been carried out for the alkali metals Na, K and Cs and for an fcc nearest neighbour Lennard-Jones model applicable to rare gas solids. The computations for the alkalis were done for several temperatures for the zero pressure zero temperature volume as well as for the zero pressure volume corresponding to each temperature. In the fcc case, results were obtained for a wide range of both the temperature and density.

Lattice dynamics calculations of the harmonic and the lowest order anharmonic (cubic and quartic) contributions to the mean square displacement were performed for the same potential models as in the molecular dynamics calculations. The Brillouin zone sums arising in the harmonic and the quartic terms were computed for very large numbers of points in q-space, and were extrapolated to obtain results fully converged with respect to the number of points in the Brillouin zone.

ABSTRACT

An excellent agreement between the molecular dynamics and lattice dynamics results was observed in the case of all the alkali metals, except for the zero pressure case of Cs, where the difference is about 15 % near the melting temperature. It was concluded that for the alkalis, the lowest order perturbation theory works well even at temperatures close to the melting temperature.

For the fcc nearest neighbour model it was found that the number of particles (256) used for the molecular dynamics calculations, produces a result which is somewhere between 10 and 20 % smaller than the value converged with respect to the number of particles. However, the general temperature dependence of the mean square displacement is the same in molecular dynamics and lattice dynamics for all temperatures at the highest densities examined, while at higher volumes and high temperatures the results diverge. This indicates the importance of the higher order (eg. λ^4) perturbation theory contributions in these cases.

ACKNOWLEDGEMENTS

It is a pleasure to thank Dr. R. C. Shukla for suggesting this project, for his guidance, valuable advice and financial assistance from his NSERC grant. I am also indebted to him for providing the values of the cubic contributions to the mean square displacement, without which the lattice dynamics calculations would have been incomplete.

I wish to acknowledge my gratitude to the Computing Centre for providing me with thousands of hours of cpu time and to the Computing Centre staff for cooperating with never ending requests and suggestions. I would especially like to thank the computer operators M. Balsom, C. Catling, R. Gare, R. Johnson and R. Slota for patiently handling stacks of printouts, mounting hundreds of tapes and running countless plots. I greatly appreciate the support and advice from M. Ramella and M. Viilma of Academic Services and the sense of humour in which they accepted dozens of reports of software bugs.

ACKNOWLEDGEMENTS

I would further like to thank the Computer Science Department for granting access to the VAX computer and the semi-intelligent terminals which saved much of my time when preparing the manuscript, and J. Sentineal for his help when I encountered problems with the VAX or RUNOFF.

Finally, my thanks are directed to A. Gerrard for patiently proof-reading the thesis.

All computations have been carried out on Brock University's Burroughs B6700 computer. The thesis was written with the help of the RUNOFF text formating program on the University's VAX 11/780 computer. Most typos were eliminated by the spelling checking program SPELL written by D. DePottie. The graphs were generated using the CalComp plot system on the B6700 and the formulae were drawn by the program FORT also using CalComp.

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

The goal of this thesis is to compute the mean square of the atomic displacement (MSD) in alkali metals and rare gas solids.

The MSD is important for several reasons: It is closely related to the Debye Waller factor (DWF), which is related to the proportion of recoilless re-absorption of gamma rays in the Mössbauer effect, as well as to the intensity of x-ray diffraction in crystals. Furthermore, the MSD determines the Van Hove perturbation expansion parameter λ, which is defined in terms of the ratio of MSD and the nearest neighbour distance (Van Hove 1961). The knowledge of λ is needed in determining the importance of the higher order contributions in perturbation theory (PT) calculations of the Helmholtz free energy or width and shift of phonons. Finally, MSD is one of the main factors in estimating the melting point of a crystal (Lindemann's melting criterion, Pines 1964).

Recently Shukla and Mountain (1982) have performed molecular dynamics (MD) and lattice dynamics (LD) computations of MSD in Li and Rb. It seems appropriate to complete this work by performing the MSD calculations for the remaining elements in the group, Na, K and Cs.

A search of the existing literature reveals that so far no comprehensive study of MSD in rare gas solids has been reported. Hence this seems a worthwhile undertaking.

Alkali metals and rare gas solids exhibit important differences. The latter crystallize in an fcc lattice which was shown by Born and Huang (1954) to form a stable structure under the nearest neighbour central force interaction. The alkalis on the other hand form a bcc lattice which does not form a stable structure under the nearest neighbour interaction (Shukla, 1981). The cores of alkali atoms are isoelectronic to rare gas atoms. The addition of one electron per ion core in the alkali system completely changes the nature of the interatomic interaction from weak and short ranged Van der Waals forces (rare gas solids) to strong and far reaching ion-ion and electron-ion interactions in alkali metals. This makes it interesting to compare the two cases.

INTRODUCTION

There are two main techniques employed in computing average properties of macroscopic systems: simulation and lattice dynamics. Calculating MSD by simulation requires the explicit knowledge of the simulated trajectories of all particles. From these, MSD can be computed in a straightforward way. Within the limits of numerical precision, simulation provides an 'exact' result. Lattice dynamics on the other hand is based on perturbation theory, the exact result is the limit of an infinite series which is truncated after a few terms.

Currently, two main simulation methods are in use: MD and Monte Carlo (MC).

In MD, Newton's equations of motion are set up using the knowledge of the interatomic forces. The equations are then solved numerically. This process naturally provides the phase space trajectories of all particles, which can then be used to compute MSD. In MC, random configurations are chosen and weighted according to their probability as given by the Boltzmann factor $\exp(-U/k_BT)$, U denoting the total potential energy (Metropolis et al, 1953). Since the displacements between two successive configurations are by definition totally random, successive configurations do not represent successive instances in time. There is no concept of time in this kind of MC simulation, no trajectories of particles are available.

A completely different procedure is the lattice dynamics (LD) method. Here each particle is thought to be located close to some equilibrium lattice site. The pair potential function is then expanded in powers of the displacement from the equilibrium position, and perturbation theory is applied to compute the harmonic as well as the anharmonic contributions to MSD. For the success of the LD method a rapid convergence of the perturbation series is essential.

In this thesis, both the MD and LD methods were used to compute the MSD in alkali metals and rare gas solids for a wide range of temperatures and volumes. A comparison of the results obtained by the two methods will shed some light on the adequacy of the lowest order perturbation theory in LD.

CHAPTER 2

MOLECULAR DYNAMICS

2.1 History of MD

The MD method was first employed by Alder and Wainwright (1959). They computed the equation of state as well as transport coefficients in systems of hard disks and spheres; their main interest was in phase transitions. The simple form of the potential function was essential given the computer power available in those days.

Rahman (1964) was the first to use a more realistic model. He computed pair correlations, self diffusion and velocity autocorrelation in liquid Ar using a two neighbour Lennard-Jones potential. Verlet (1967) who worked on the same Lennard-Jones model of liquid Ar introduced a programming technique which he called "bookkeeping". This significantly reduced the computer time consumption for simulations with short ranged potentials.

2.2 General Description

The main advantages of MD are as follows: Only very few a priori assumptions are made. This makes the method less dependent on theoretical models; MD is in fact a 'computer experiment' which allows to test different theoretical models. The advantage is that MD simulations can provide information not directly accessible by 'real' experiments, for example trajectories of individual particles. MD even allows to 'experiment' with idealized or unphysical systems. Unlike Monte Carlo methods, MD also allows the computation of dynamic properties. The method's main disadvantage is the high requirement of computing time. Usually the simulated object is in a liquid or solid state, especially well suited to MD simulations are crystals due to their regular lattice structure.

As mentioned earlier, MD numerically solves the classical equations of motion of an N particle system. The equations of motion for an N particle system are given by

$$M_i \ddot{\vec{r}}_i = -\sum_{j=1}^{N}{}' \frac{\partial \Phi(r_{ij})}{\partial \vec{r}_{ij}} \qquad (2.1)$$

where M is the mass of the i-th particle located at $\vec{r}_i$, $\Phi_{ij}(r)$ is the pair interaction potential between particles i and j, $\vec{r}_{ij} := \vec{r}_j - \vec{r}_i$ and the prime on the summation sign indicates omission of the term j=i. They are a system of N linear

differential equations, second order in the time derivatives, coupled by the pair potential. The numerical integration is done by first transforming Eq. (2.1) into difference equations involving a finite time step Δt. Given the coordinates $\vec{r}_i$, the velocities $\vec{v}_i := \dot{\vec{r}}_i$, the accelerations $\vec{a}_i := \ddot{\vec{r}}_i$ and possibly higher derivatives for some time t (and possibly earlier times), the coordinates and their derivatives can then be obtained for time $t+\Delta t$.

Since the amount of computer time required for the simulation increases at least linearly with the number of particles simulated, this number is usually limited to several hundred, or at most, a few thousand. In order to simulate a macroscopic system consisting of some 10^{23} particles, periodic boundary conditions are imposed. The system that is actually dealt with is therefore an infinite array of identical cells, each containing N particles. If in the course of the simulation a particle leaves its cell another particle is brought in, thus keeping the number of particles in the cell constant.

2.3 Basic Assumptions

As mentioned earlier, one main advantage of MD is the small number of a priori assumptions made. The following is a discussion of these assumptions.

- Classical system:

  It is assumed that the particles obey the classical equations of motion. This limits MD simulations to temperatures high enough to ignore quantum effects.

- Pair potential:

  Only two body forces are considered. This is not an inherent restriction of MD. In principle three and more body force models could be used at the expense of a huge increase in computer time requirements. However in most cases two body forces give satisfactory results.

- Finite range of the pair potential:

  Only the contributions of a finite number of particles can be examined to calculate the total force acting on a given particle. This is justified by the fact that realistic pair potentials usually die out quickly. It is well known that in solids only a few shells of neighbours interact considerably with any particle. The contributions from particles further away cancel due to screening effects of other particles. The range of the potential can also be part of the 'experimental setting': It is possible to study

the effect of limiting the range of the interaction, which of course can't be done in real experiments.

- Small number of particles (much less than 10^{23}) with periodic boundary conditions simulate a macroscopic system:

  As explained above, this is dictated by the finite computing power available and implies the simulation of an infinite, but strictly periodic system. This means that the number of degrees of freedom of the simulated system is very small compared to a real macroscopic system. Therefore the MD simulation will not produce effects arising due to the collective motion of a large number of particles. It also implies the absence of surface effects, unless the simulation is explicitly set up for the study of surfaces (semi-infinite system), in which case most of the bulk properties will be lost. The justification for using small numbers of particles lies in the fact that due to the finite range of the potential, particles far apart cannot interact directly.

- Choice of the potential:

  The choice of the potential depends on the purpose of the simulation. For example, the potential may be obtained from a theoretical model and the simulation can be used to produce results which can be checked against experimental data. Alternatively, quantities which cannot be measured directly may be computed using first principles type potentials or potentials fitted to experimental data.

2.4 Fundamental Decisions Regarding the Simulation

This section will investigate in a general way the main choices which have to be made before a MD simulation can be performed.

2.4.1 The Integration Algorithm

The integration algorithm provides a way to compute the particles' coordinates and their time derivatives for time t+ t given their values at time t (and possibly earlier times).

The most straightforward method might be to first obtain the accelerations a (t) as

$$\vec{a}_i(t) := -\frac{1}{M_i}\sum_{j=1}^{N}{}' \frac{\partial \Phi_{ij}(r_{ij}(t))}{\partial \vec{r}_{ij}(t)} \qquad (2.2)$$

next to determine the velocities as

$$\vec{v}_i(t+\Delta t) := \vec{v}_i(t) + \vec{a}_i(t)\Delta t \qquad (2.3)$$

and finally the positions from

$$\vec{r}_i(t+\Delta t) := \vec{r}_i(t) + \frac{1}{2}[\vec{v}_i(t) + \vec{v}_i(t+\Delta t)]\Delta t \qquad (2.4)$$

However, this very simple algorithm doesn't perform too well, the algorithm due to Beeman exhibits a better numerical behaviour (Sangster and Dixon 1976). Here the accelerations from times t and t-Δt are first used to determine the new coordinates as

$$\vec{r}_i(t+\Delta t) := \vec{r}_i(t) + \vec{v}_i(t)\Delta t + \frac{1}{6}[4\vec{a}_i(t) - \vec{a}_i(t-\Delta t)](\Delta t)^2 \quad (2.5)$$

From these the new accelerations are computed according to (2.2). Finally the new velocities are obtained as

$$\vec{v}_i(t+\Delta t) := \vec{v}_i(t) + \frac{1}{6}[2\vec{a}_i(t+\Delta t) + 5\vec{a}_i(t) - \vec{a}_i(t-\Delta t)]\Delta t \quad (2.6)$$

The Beeman algorithm requires storing all accelerations from three successive time steps. This disadvantage is balanced by a better energy conservation, which allows the use of a larger time increment Δt.

Also popular are predictor-corrector algorithms. They first use the coordinates and their derivatives at time t to predict the coordinates $\vec{r}_i'(t+\Delta t)$. These are then used to determine predicted accelerations $\vec{a}_i'(t+\Delta t)$, which are in turn used to correct the positions $\vec{r}_i(t+\Delta t)$. These are finally used to correct the accelerations obtaining $\vec{a}_i(t+\Delta t)$. The correction can be repeated until a required precision is achieved. An example is the algorithm used by Rahman (1964). The equations are:

$$\vec{r}_i'(t+\Delta t) := \vec{r}_i(t-\Delta t) + 2\vec{v}_i(t)\Delta t \quad (2.7)$$

$$\vec{v}_i(t+\Delta t) := \vec{v}_i(t) + \frac{1}{2}[\vec{a}_i'(t+\Delta t) + \vec{a}_i'(t)]\Delta t \qquad (2.8)$$

$$\vec{r}_i(t+\Delta t) := \vec{r}_i(t) + \frac{1}{2}[\vec{v}_i(t+\Delta t) + \vec{v}_i(t)]\Delta t \qquad (2.9)$$

Since the computation of the accelerations is by far the most time consuming operation, one time step of a predictor-corrector algorithm typically requires twice the computer time it takes for a single step algorithm. This means that the usage of a predictor-corrector algorithm is advisable only if the time step can at least be doubled, or if high demands are put on the absolute accuracy of the computed coordinates.

2.4.2 The Potential

The choice of the potential function was already discussed in Sect. 2.3. It remains to be pointed out that the range of the potential is somehow critical. This is due to the fact that the computer time required per time step is approximately proportional to the cube of the potential range. The range also sets a lower limit on the number of particles used (cf. Sect. 2.4.3 and 2.5.2.2).

2.4.3 The Size of the Simulation

As will be discussed in some detail later on (Sect. 2.5.2.2), the shortest translation vector which will map the simulated 'box' on itself (due to periodic boundary conditions) should be longer than twice the range of the potential. This puts a lower limit on the size of the simulation. In most cases the box will be cubic, which means that its side length should be larger than twice the potential range. For example a simulation of a bcc crystal lattice with a potential ranging out to six neighbour shells will require a box length of at least five times the lattice constant. This means $5^3=125$ conventional unit cells or N=250 particles in the simulation.

The results of the simulation will in general depend on the number of particles. In order to obtain results close to the macroscopic values (corresponding to $N=10^{23}$), a much larger N than the minimum determined by the potential range may be required.

2.4.4 Time Step and Simulated Time Interval

The algorithms discussed above produce approximate solutions of the equations of motion. Naturally, the quality of the approximation is strongly dependent on the time step Δt. Decreasing the time step increases both the accuracy and the amount of computer time required for the simulation, so that some compromise between accuracy and cost has to be found. The main indications for an appropriate choice of Δt will be energy stability and stability of the calculated results with respect to changes in Δt.

As a unit of time a typical time constant of the system might be chosen as

$$\tau = r_0 \sqrt{\frac{M}{\varepsilon}} \qquad (2.10)$$

where r_0 is some typical distance (like nearest neighbour distance or lattice constant), ε some typical energy (like the depth of the potential well) and M is the average mass of the simulated particles. In this time scale an appropriate value for Δt will typically be in the range 0.001 to 0.01.

The length t_o of the simulated time interval depends strongly on the purpose of the simulation. It must be chosen large enough to allow the 'measured' quantity to converge to some stable value. For MSD calculations a t_o of the order of 10 will generally suffice, however other calculations (like specific heat or structure factors) may require a much larger number of time steps.

2.5 Course of the Simulation

2.5.1 Initialization

2.5.1.1 Initial Coordinates

In order to start the simulation, the particles must be given some initial positions and velocities. There are several possible ways to do this. An obvious choice, especially for simulation of solids is to start all particles on equilibrium lattice sites and supply them with random velocities. Alternatively the particles can be given some random displacements from the equilibrium positions and zero or random velocities. Which one of these methods is chosen is not essential.

A different initialization scheme is to start with randomized positions. This is not appropriate for solids but may be the easiest way to start simulations of a liquid. However some particles may obtain positions very close to each other, resulting in a high potential energy. Care must be taken to avoid this situation.

Once the positions and velocities are defined, a potentially non-zero linear or angular momentum must be eliminated. This is done by adding an appropriate constant (linear or angular velocity component) to every particle.

2.5.1.2 Reaching Thermal Equilibrium

After the coordinates have been initialized, it is necessary to get rid of the influence of the arbitrary initial conditions and to get the system into thermal equilibrium at some desired temperature. The temperature of the simulated system is the average kinetic energy of the particles measured in units of the Boltzmann constant k_B. Thus, if the kinetic energy is E, the temperature is $T=2E/3Nk_B$.

The temperature can be controlled via the velocities. Depending on the initialization scheme chosen, the temperature might drop or increase dramatically during the first few time steps and will then slowly converge towards the equilibrium value. This is due to a redistribution of potential and kinetic energy on the way to thermal equilibrium. Generally, the equilibrium temperature will not be the desired one. It is therefore necessary to 'heat' or 'cool' the system by scaling all particles' velocities by a constant factor. After such a scaling process, the system will again redistribute its kinetic and potential energy, moving its temperature closer to the value it had before scaling the velocities. Therefore it will be necessary to repeat the scaling and subsequent running until the desired equilibrium temperature is reached.

The scaling factor is limited to be greater than or equal to zero for cooling. In the case of solids, the scaling factor cannot be too large, because a value much larger than one might give some particles very high energies which may lead to the destruction of the lattice structure. After scaling with a factor much different from unity it will take a long time before the system reaches the equilibrium state again. This is especially the case for low temperatures, since the coupling between the motions of different particles is weaker at lower temperatures.

To speed up this process it is generally a good idea to use a relatively large time step in the beginning, until the system is in equilibrium. The time step can then be reduced and the system is run for some more time to prepare for the production runs.

The estimate of the specific heat C as obtained from the temperature fluctuations can be used as an indication of reaching the equilibrium state. It is given as (Lebowitz et al. 1967)

$$C_v = \frac{3 k_B}{2 - 3N \frac{\langle (\Delta T)^2 \rangle}{\langle T \rangle^2}} \qquad (2.11)$$

where the angular brackets indicate averaging over a number of time steps. In equilibrium, this average should be reasonably stable and close to independent estimates like the classical value of $3k_B$. However, it is difficult to obtain a reasonable value of C_v in this fashion, and not too much stability should be expected.

2.5.2 Integration Process

2.5.2.1 Calculation of Forces

During the MD simulation most of the computing time goes into the computation of the forces acting on the particles. The total force on particle i is given as

$$\vec{F}_i = \sum_{j=1}^{N}{}' \vec{f}_{ij} = -\sum_{j=1}^{N}{}' \frac{\Phi'_{ij}(r_{ij})}{r_{ij}} \vec{r}_{ij} \qquad (2.12)$$

where $\vec{f}_{ij}$ denotes the force exerted by the particle j on the particle i. $\vec{r}_{ij}$ are subject to periodic boundary conditions (PBC). Since $\vec{f}_{ij} = -\vec{f}_{ji}$, only half of these need to be calculated.

Determination of the forces can be simplified if $\Phi'_{ij}(r)/r$ is evaluated as a function of r^2. This avoids explicit calculation of r which would require taking a square root.

A further reduction of computational effort is possible due to Verlet's (1967) "bookkeeping" procedure: If the potential is cut off at a distance r_c less than half of the maximum distance between two particles, most of the $\vec{f}_{ij}$ will be zero. To avoid computing these non-contributing terms, a list can be set up of all pairs of particles less than a distance $r_k > r_c$ apart. For the next k time steps, only pairs from this list are considered when calculating the forces. r_k must be chosen so large that no pair of particles, which is not in the aforementioned list, can get closer than r_c during the next k time steps. If this condition

is always satisfied, no error will be made by ignoring all pairs not in the list. r_k can be determined as

$$r_k \stackrel{a}{=} r_c + bk\langle v\rangle\Delta t \tag{2.13}$$

where the 'safety factor' b is of the order of unity and must be adjusted in a particular case to ensure the above condition to hold.

2.5.2.2 Periodic Boundary Conditions

The periodic boundary conditions enter in the determination of the distance vector $\vec{r}_{ij}$ as well as the new positions $\vec{r}_i(t+\Delta t)$.

Provided that the potential is short ranged enough to allow any particle i to interact with at most one of particle j's mirror particles (including j itself), $\vec{r}_{ij}$ must be the vector pointing from i's position to the position of the nearest mirror particle of j. Mathematically this means that

$$r_{ij} \stackrel{a}{=} \min_{\tau\in\Theta}(|\vec{r}_j - \vec{r}_i + \tau|) \tag{2.14}$$

where Θ is the set of all translations leaving the simulated system invariant. If the simulated 'box' is rectangular with side lengths l_α, then

$$|r_{ij}^{(\alpha)}| \stackrel{a}{=} \min_{s=-1,0,1}(|r_j^{(\alpha)} - r_i^{(\alpha)} + s l_\alpha|) \tag{2.15}$$

or

$$r_{ij}^{(\alpha)} \stackrel{a}{=} \begin{cases} d_{ij}^{(\alpha)} & \text{if } |d_{ij}^{(\alpha)}| \geq \frac{1}{2} l_\alpha \\ d_{ij}^{(\alpha)} - l_\alpha \operatorname{sign}(d_{ij}^{(\alpha)}) & \text{otherwise} \end{cases} \tag{2.16}$$

where $r_i^{(\alpha)}$ designates the α-th cartesian component of $\vec{r}_i$ and $\vec{d}_{ij} := \vec{r}_j - \vec{r}_i$.

If the side length of the box is smaller than twice the potential range, particle i can interact with more than one of j's mirror particles. This is a very unphysical effect of the PBC, to ignore it seems hard to justify, especially in solids. The number of particles should therefore be chosen large enough to avoid this situation.

PBC must be applied again when the new positions are calculated using the integration algorithm. If the new position $\vec{r}_i'(t+\Delta t)$ happens to lie outside the 'box', a suitable translation must be applied to move it back in. If the 'box' is rectangular and symmetrical around the origin, the new position is given as

$$r_i^{(\alpha)} \stackrel{a}{=} \begin{cases} r_i'^{(\alpha)} & \text{if } |r'^{(\alpha)}| \leq \frac{1}{2} l_\alpha \\ r_i'^{(\alpha)} - l_\alpha \operatorname{sign}(r_i'^{(\alpha)}) & \text{otherwise} \end{cases} \tag{2.17}$$

CHAPTER 3

MEAN SQUARE DISPLACEMENT

3.1 Discussion of the Models Used

3.1.1 Alkali Metals

The potential for the alkali metals was the same as used by Shukla and Mountain (1982), constructed according to Price et al. (1970). The potential is based on a self consistent screening theory of electron correlations (Vashishta and Singwi, 1972) and incorporates electron ion interactions by means of an Ashcroft pseudopotential (Ashcroft 1966). The potential parameters are listed in tables 3.1 and 3.2. In accordance with Shukla and Mountain it was truncated after the sixth neighbour distance.

During the calculations all quantities were expressed in dimensionless units. The basic units were as follows: for length the lattice constant a, for energy the minimum value ϵ of the potential function and for mass the atomic mass M. These values and the time unit τ derived from them according to (2.10) are also listed in tables 3.1 and 3.2.

Table 3.1: Volume independent potential parameters for alkali metals

Element	M [amu]	M [10^{-27} kg]	M^*/M	r_c/a_o
Na	22.990	38.18	1.0	1.69
K	39.098	64.92	0.93	2.226
Cs	132.905	220.69	0.86	2.62

Table 3.2: Volume dependent potential parameters for alkali metals†

Element	a [Å]	r_s^*/a_o	A	B	ε [10^{-21} J]	τ [10^{-12} s]
	4.225	3.931	0.995	0.263	5.939	1.071
	4.234	3.940	0.994	0.263	5.978	1.070
Na	4.251	3.955	0.995	0.262	6.019	1.071
	4.262	3.966	0.995	0.262	6.045	1.071
	4.288	3.990	0.996	0.262	6.109	1.072
	4.309	4.009	0.996	0.261	6.162	1.073
	5.225	4.519	1.007	0.249	5.647	1.772
	5.261	4.552	1.007	0.248	5.662	1.781
K	5.277	4.566	1.007	0.247	5.681	1.784
	5.305	4.591	1.008	0.247	5.712	1.789
	5.343	4.623	1.008	0.246	5.756	1.794
	6.045	4.840	1.011	0.242	5.235	3.925
	6.069	4.856	1.012	0.241	5.225	3.944
Cs	6.092	4.875	1.012	0.240	5.240	3.954
	6.119	4.898	1.012	0.240	5.259	3.964
	6.163	4.932	1.013	0.239	5.290	3.981

a_o is the Bohr radius

† The lattice constants for Na and K were taken from Shukla and Taylor (1974), those for Cs from Shukla and Plint (1982)

3.1.2 Rare Gas Solids

A nearest neighbour Lennard-Jones potential

$$\Phi(r) = 4\varepsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right] \tag{3.1}$$

was chosen for the rare gases. ε and σ were used as units for energy and length respectively. Since ε is scaled out in all calculations, the only free parameter is σ, the root of the potential function. The minimum of this potential function is at $r_0 = \sqrt[6]{2}\,\sigma$. The potential parameters are usually fitted for zero temperature and pressure. That means that at 0 K r_0 will be equal to the nearest neighbour distance r_1. At higher temperatures r_1 will be larger than r_0 due to thermal expansion. It is convenient to characterize the volume dependence by a dimensionless parameter a_1 which incorporates the slope and curvature of the potential function at the nearest neighbour distance. This parameter can be defined as

$$a_1 = \frac{\Phi'(r_1)}{r_1\Phi''(r_1) - \Phi'(r_1)} \tag{3.2}$$

(Shukla and MacDonald 1980, Shukla 1980). Substituting (3.1) into (3.2) yields

$$\frac{r_1}{\sigma} = \sqrt[6]{\frac{2 + 28a_1}{1 + 8a_1}} \tag{3.3}$$

for the Lennard-Jones potential. A value of $a_1 = 0$ corresponds to

the zero pressure zero temperature volume, while at higher temperatures the zero pressure volume will correspond to a positive value of a_1. Negative a_1-values correspond to non-zero pressure and also occur in all neighbour Lennard-Jones potentials sometimes used for fcc crystals. Our calculations were carried out for a_1-values between -0.04 and +0.1 in steps of 0.02.

3.2 Molecular Dynamics Calculations

3.2.1 Procedure

MSD is the ensemble average of the square of the displacement of particles from their equilibrium position. In MD this average is taken over both, ensemble and time:

$$\langle(\Delta r)^2\rangle = \frac{1}{t_0}\int_0^{t_0}\frac{1}{N}\sum_{i=1}^{N}[\vec{r}_i(t)-\langle\vec{r}_i\rangle]^2\,dt \qquad (3.4)$$

This can be written as

$$\langle(\Delta r)^2\rangle = \frac{1}{N}\sum_{i=1}^{N}[\langle r_i^2\rangle - \langle\vec{r}_i\rangle^2] \qquad (3.5)$$

where the averages on the RHS are sums over $n=t_0/\Delta t$ time steps:

$$\langle\vec{r}_i\rangle := \frac{1}{n}\sum_{\nu=1}^{n}\vec{r}_{i\nu}\,, \qquad \langle\vec{r}_i^2\rangle := \frac{1}{n}\sum_{\nu=1}^{n}r_{i\nu}^2 \qquad (3.6)$$

and $\vec{r}_{i\nu} := \vec{r}_i(\nu\,\Delta t)$.

The procedure is to accumulate during the simulation for every particle the sums of the $\vec{r}_{i\nu}$ and $r^2_{i\nu}$. At the end MSD can then be computed according to (3.5) and (3.6).

3.2.2 Integration Algorithm and Parameters

As in the work by Shukla and Mountain (1982) the Beeman algorithm discussed in Sect. 2.4.1 was chosen to integrate the equations of motion.

The simulations of the alkali metals were carried out for N=250 particles, corresponding to a cubic box with a side length l=5a, where a is the lattice constant. The time step was $\Delta t = 0.002\tau$ and the averages were computed over a period of 3000 time steps. With this choice the fluctuations of the total energy were of the order of 20 ppm (parts per million) and an average loss of energy of 2 ppm was recorded per time step.

The fcc calculations were done for N=256 particles (l=4a), except for one run using n=500 (l=5a). The time step was $\Delta t = 0.001\tau$ and the integrations were carried out for 6000 time steps. This produced energy fluctuations of the order of 10 ppm and an average loss of energy of 2 ppm per time step.

3.3 Lattice Dynamics Calculations

The expressions for the various contributions to the DWF in the high temperature (HT) limit have been derived by Maradudin and Flinn (1963). Since the MSD expressions are closely related to DWF, the quasiharmonic and anharmonic $O(\lambda^2)$ contribution are given by

$$\langle(\Delta r^2)\rangle_{QH} = \frac{k_B T}{NM}\sum_{\vec{q}j}\frac{1}{\omega^2(\vec{q}j)} \tag{3.7}$$

$$\langle(\Delta r)^2\rangle_1 = -\frac{(k_B T)^2}{2N^2M^3}\sum_{\vec{q}_1\vec{q}_2}\sum_{j_1j_2}\frac{\Phi(\vec{q}_1j_1,-\vec{q}_1j_1,\vec{q}_1j_1,-\vec{q}_2j_2)}{\omega^2(\vec{q}_1j_1)\omega^4(\vec{q}_2j_2)} \tag{3.8}$$

and

$$\langle(\Delta r)^2\rangle_2 = \frac{(k_B T)^3}{2N^2M}\sum_{\vec{q}_1\vec{q}_2\vec{q}_3}\sum_{j_1j_2j_3}\Delta(\vec{q}_1+\vec{q}_2+\vec{q}_3) \tag{3.9}$$

$$\times\ \frac{\Phi(\vec{q}_1j_1,\vec{q}_2j_2,\vec{q}_3j_3)\Phi(-\vec{q}_1j_1,-\vec{q}_2j_2,-\vec{q}_3j_3)}{\omega^2(\vec{q}_1j_1)\omega^2(\vec{q}_2j_2)\omega^2(\vec{q}_2j_2)}$$

where $\omega(\vec{q}j)$ is the phonon frequency of the wave vector $\vec{q}$ and branch index j and the Φ functions represent the Fourier transforms of the third an fourth rank tensor derivatives of the potential function.

According to Shukla and Mountain (1982), the first of the anharmonic contributions (quartic term) can be written as

$$\langle(\Delta r)^2\rangle_1 = -\frac{(k_B T)^2}{N^2 M^3}\sum_{\vec{l}}{}'' \sum_{\alpha\beta\gamma\delta} \Phi_{\alpha\beta\gamma\delta}(\vec{l}) \times \left[S_{\alpha\beta}(0)-S_{\alpha\beta}(\vec{l})\right]\left[T_{\gamma\delta}(0)-T_{\gamma\delta}(\vec{l})\right] \tag{3.10}$$

where $\Phi_{\alpha\beta\gamma\delta}(\vec{l})$ are the fourth partial derivatives of the potential function evaluated for the direct lattice vector $\vec{r}_{\vec{l}}$ and the tensors S and T are defined as

$$S_{\alpha\beta}(\vec{l}) = \sum_{\vec{q}j} \frac{e_\alpha(\vec{q}j)e_\beta(\vec{q}j)}{\omega^2(\vec{q}j)}\cos(\vec{q}\cdot\vec{r}_{\vec{l}}) \tag{3.11}$$

$$T_{\alpha\beta}(\vec{l}) = \sum_{\vec{q}j} \frac{e_\alpha(\vec{q}j)e_\beta(\vec{q}j)}{\omega^4(\vec{q}j)}\cos(\vec{q}\cdot\vec{r}_{\vec{l}}) \tag{3.12}$$

where $\vec{e}(\vec{q}j)$ is the eigenvector of the dynamical matrix whose eigenvalue is $\omega^2(\vec{q}j)$.

In order to calculate the quasiharmonic and cubic contributions, the dynamical matrix is diagonalized for a grid of points in q-space in the irreducible 1/48-th sector of the Brillouin zone (BZ). The resulting eigenvectors and eigenvalues are then used to compute the BZ sums arising in (3.7), (3.11) and (3.12), after transforming the whole BZ sums to 1/48-th of the BZ by the procedure given in Shukla and Wilk (1974).

The values of the BZ sums depend on the number of wave vectors used. The convergence with increasing number of wave vectors is not very good. Since the largest contributions arise from the vicinity of the origin, using a shifted mesh of points rather than a simple cubic mesh improves the results. This is because a shifted mesh will produce points closer to the origin than does a simple mesh with the same density. For the calculations of the harmonic part, the results were improved further by using a non-uniform mesh with a higher density of points in the vicinity of the origin.

The sums were computed for a steplength L as high as 100 (85,850 vectors in the irreducible sector of the BZ). From this, an extrapolated value for $L=\infty$ was obtained by linear regression of the values of the sums as a function of $1/L$.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Reliability of the MD Results

The reliability of the MD results depends on three parameters: the time increment, Δt, the number of integration steps, n, and the number of simulated particles, N. The influence of the first two of these parameters can be easily determined by comparing runs with different parameter settings. The N dependence is much harder to assess and will be dealt with later on.

The effect of the parameters n and Δt was explored in several runs using the values 0.01, 0.005 and 0.002 for Δt and up to 6000 integration steps (n). The results are summarized in Table 4.1.

Table 4.1: Effects of n and Δt on the MD result for MSD

time step Δt	0.002	0.005	0.01
final MSD value	0.0234	0.0232	0.0224
n required for stable MSD	2000-3000	1000-2000	1000-2000
stability of MSD value	1 %	2 %	4 %
energy loss per 1000 time steps	0.6 %	2 %	2.5 %
C_v estimate	2.5	never stabilized	

RESULTS AND DISCUSSION

It can be seen that with the longer time steps the energy loss is drastically increased compared to the case $\Delta t=0.002$. Moreover, the C_v estimate never stabilized near a value of 3 during the whole simulation. From these tests it was concluded that the choices $\Delta t=0.002$ and $n=3000$ would produce results converged to about 1 % with respect to n and probably 1 to 2 % with respect to Δt. The former conclusion was confirmed when comparing results of subsequent runs, where the final configuration of one simulation was taken as the initial configuration for the next run. The results generally differed by less than 1 %.

The errors quoted in the tables summarizing the MD results are estimated from the fluctuations during the last 1000 time steps. They do not reflect the error introduced by the choices of Δt and N. In the case of the fcc simulations, 5000 to 6000 time steps were required to stabilize the MSD results to less than 1 % for the chosen time increment $\Delta t=0.002$. The convergence with respect to Δt is estimated to be within 2 to 4 %.

The dependence of the simulation results on N can be investigated in two ways: Simulations can be run using different numbers of particles, and the MD results can be compared with LD for lower temperatures, where the higher order PT contributions are exceedingly small. The first possibility is far more complicated and time consuming than the aforementioned comparisons between different Δt and n values. This is because a simulation using a different number of particles must always start from scratch, while Δt and n can be changed during the

simulation. For this reason, only one such comparison was made: For the fcc nearest-neighbour Lennard-Jones model, the MSD calculation for $a_1 = 0.00$, $k_B T/\varepsilon = 0.48$ was done for N=256 and for N=500. The results in reduced units were MSD/T = 0.07030 for N=256 and MSD/T = 0.07628 for N=500, the latter being about 8.5 % larger. This indicates that the converged result (with respect to N) would be at least 10 % higher than the value for N=256.

This observation is confirmed by the comparisons with LD. At the lowest temperature ($k_B T/\varepsilon = 0.12$) the anharmonic contribution to MSD at $a_1 = 0.0$ is only 2.5 % of the harmonic value. It is safe to assume that any anharmonic contributions of $O(\lambda^4)$ would be much less than 1 %. The LD value should therefore be within less than 1 % of the fully converged answer. However the MD result in this case differs from the LD value by 16 %. This discrepancy can be explained as the effect of the finite value of N and it can serve as an estimate of the total error of the MD calculations. At the highest a_1-ratio this difference is only about 10 %, while at $a_1 = -0.04$ it is as large as 20 %.

Although no runs were made to examine the N-dependence of the MSD results for alkali metals, some indication can be given here based on the rare gas results. Here the difference between MD and LD at the lowest temperatures used is between 4.5 and 9 %, indicating that running with a larger number of particles will not produce any substantially different results.

4.2 Reliability of the LD Results

As explained earlier, fully converged BZ sums were used to determine the quasiharmonic and the quartic contributions to MSD. This means that the errors in the quasiharmonic and quartic terms will be 100 ppm or less. The cubic contribution to MSD was calculated using a mesh of 432 wave vectors in the whole BZ for the bcc case and 4000 vectors for the fcc runs. The errors were estimated to be approximately 10 % and 5 % respectively.

As one check of the LD calculation, the quasiharmonic and quartic contributions were computed for Rb, a = 5.739 Å. The fully converged answers were $S_{QH} = 0.42370\ m^2/J$ and $S_1 = -2.0999 \times 10^{19}\ m^2/J^2$, as compared to $S_{QH} = 0.41014\ m^2/J$ and $S_1 = -1.9631 \times 10^{19}\ m^2/J^2$ as obtained by Shukla and Mountain (1982). The difference is due to our usage of fully converged $S_{\alpha\beta}$ and $T_{\alpha\beta}$ tensors.

The only other calculation of MSD in rare gas solids, which we can compare with our MD and LD results, is that of Goldman (1968), who used a highly approximate frequency shift analysis of MSD. His result for MSD in Xe using the zero pressure zero temperature volume at a temperature of 160 K (that is a reduced temperature of 0.483), is 0.0092 in our units. (This number was obtained from the graph provided in his paper). Our harmonic result for this case is 0.0107, with the anharmonic terms added this reduces to 0.0098. Since Goldman did not calculate the $O(\lambda^2)$ contributions to MSD and the precision with which the

graphs in his paper can be read is very limited, a better agreement cannot be expected.

Our MD results with 256 and 500 particles respectively are 0.0085 and 0.0093 for this case. It can be seen that while the number for 256 particles differs from both, Goldman's result and our LD value, the usage of 500 particles improves the MD result considerably.

4.3 Results for Alkali Metals

The converged tensors used in the LD calculations of MSD are presented in Appendix I. They are made dimensionless by multiplying with α_1 defined as

$$\alpha_1 = \frac{1}{3}\left[\frac{2}{r_1}\Phi'(r_1) + \Phi''(r_1)\right] \tag{4.1}$$

α_1 is also given in Appendix I.

In Table 4.2 the quasiharmonic and anharmonic contributions are given in terms of S_{QH}, S_1 and S_2, which are defined as

$$3(k_B T)S_{QH} = \langle(\Delta r)^2\rangle_{QH} \tag{4.2}$$

$$3(k_B T)^2 S_1 = \langle(\Delta r)^2\rangle_1 \tag{4.3}$$

and

$$3(k_B T)^2 S_2 = \langle (\Delta r)^2 \rangle_2 \qquad (4.4)$$

The values of the cubic contributions were provided by Shukla (1984).

In Table 4.3 the MD results are presented and compared with LD. A graphical representation of the MD and LD results is given in Fig. 4.1.

As can be seen from Table 4.3, the MD results at the lowest temperatures ($T \sim \Theta_D$, where Θ_D is the Debye- Temperature) are generally about 3 to 5 % lower than the quasiharmonic contribution. At higher temperatures MD exceeds the quasiharmonic value by 5 to 25 %. The anharmonic contributions are all positive. At lower temperatures anharmonicity contributes some 2 % while at the highest temperatures ($T \sim T_m$, where T_m is the melting temperature) they increase the quasiharmonic value by about 10 to 12 %. Adding the anharmonic terms generally increases the difference between MD and LD at the lower temperatures while improving the agreement at higher temperatures. As explained earlier, the 4.5 to 9 % difference at lower temperatures can be attributed to the finite value of N in the MD calculations. The agreement at higher temperatures is generally excellent (1 to 5 %) with the exception of the zero pressure MSD result for Cs. In this case the LD and MD results differ by 15 % at the highest temperature. However we note that

without taking into account the anharmonic contributions, the difference is about 27 %.

It must be pointed out that, while the quasiharmonic and quartic contributions were computed to very high accuracy, the cubic terms are only converged to approximately 10 %. Since the quartic contributions are about 60 to 80 % of the cubic values but with opposite sign, the fully converged answer for the total of the $O(T^2)$ terms could be in some cases up to 50 % higher than calculated. This could make the agreement between molecular dynamics and lattice dynamics almost perfect for Na and reduce the difference greatly in the case of Cs. The remaining differences between MD and LD calculations especially in Cs indicate small but visible contributions from $O(T^3)$ terms.

Table 4.2: LD results for MSD in Na, K and Cs

Element	a	S_{QH}	S_1	S_2
Na	4.225	0.2190	-16.03	20.47
	4.234	0.2228	-16.37	21.34
	4.251	0.2302	-18.97	23.15
	4.262	0.2351	-18.66	24.41
	4.288	0.2474	-23.27	27.84
	4.309	0.2582	-25.61	31.10
K	5.225	0.3220	-14.33	22.64
	5.261	0.3381	-16.16	25.15
	5.277	0.3455	-17.50	26.39
	5.305	0.3592	-18.88	28.76
	5.343	0.3788	-21.95	32.46
Cs	6.045	0.4664	-20.57	33.65
	6.069	0.4790	-22.27	35.65
	6.092	0.4919	-24.10	37.72
	6.119	0.5075	-26.46	40.36
	6.163	0.5347	-30.57	45.25

Units are Å for a, m^2/J for S_{QH} and 10^{18} m^2/J^2 for S_1 and S_2.

Table 4.3: MD results for MSD in Na, K and Cs and comparison with LD

Element	a	T	QH	AH	QH+AH	MD
Na	4.225	93 (4)	8.44	0.22	8.66	7.98(1)
	4.234	93 (3)	8.58	0.25	8.83	8.30(5)
	4.225	163 (7)	14.79	0.67	15.46	14.68(4)
	4.251	163 (7)	15.54	0.68	16.22	16.11(8)
	4.225	227 (8)	20.59	1.31	21.90	21.69(10)
	4.262	225 (9)	21.91	1.67	23.57	21.31(10)
	4.225	282 (9)	25.58	2.02	27.60	26.98(20)
	4.288	294(14)	30.13	2.26	32.38	32.82(10)
	4.225	377(12)	34.20	3.61	37.81	40.07(30)
	4.309	365(14)	39.03	4.26	43.29	45.42(10)
K	5.225	102 (4)	13.60	0.49	14.10	12.90(10)
	5.261	101 (4)	14.14	0.52	14.67	13.22(10)
	5.225	164 (6)	21.87	1.28	23.15	22.47(20)
	5.277	163 (5)	23.33	1.35	24.68	22.94(10)
	5.225	218 (8)	29.07	2.26	31.33	29.39(10)
	5.305	215 (8)	31.99	2.61	34.60	31.29(30)
	5.225	282(11)	37.61	3.78	41.39	39.85(10)
	5.343	291(10)	45.66	5.09	50.75	*51.24(4)
Cs	6.045	103 (4)	19.90	0.79	20.69	19.78(70)
	6.069	103 (4)	20.43	0.81	21.25	19.69(5)
	6.045	161 (7)	31.10	1.94	33.04	30.93(70)
	6.092	160 (6)	32.60	1.99	34.59	31.70(15)
	6.045	219 (9)	42.31	3.59	45.89	46.10(30)
	6.119	218 (9)	45.82	3.78	49.60	46.08(30)
	6.045	262(10)	50.61	5.13	55.75	*54.99(30)
	6.163	293(10)	64.89	7.21	72.10	82.51(30)

a is given in Å, T in K, MSD in 10^{-22} m^2.
A value in parentheses gives the error of the preceding value in units of the last digit. QH and AH are the quasiharmonic and λ^2 anharmonic contributions to MSD. MD results marked with an asterisk are obtained using n=6000 time steps.

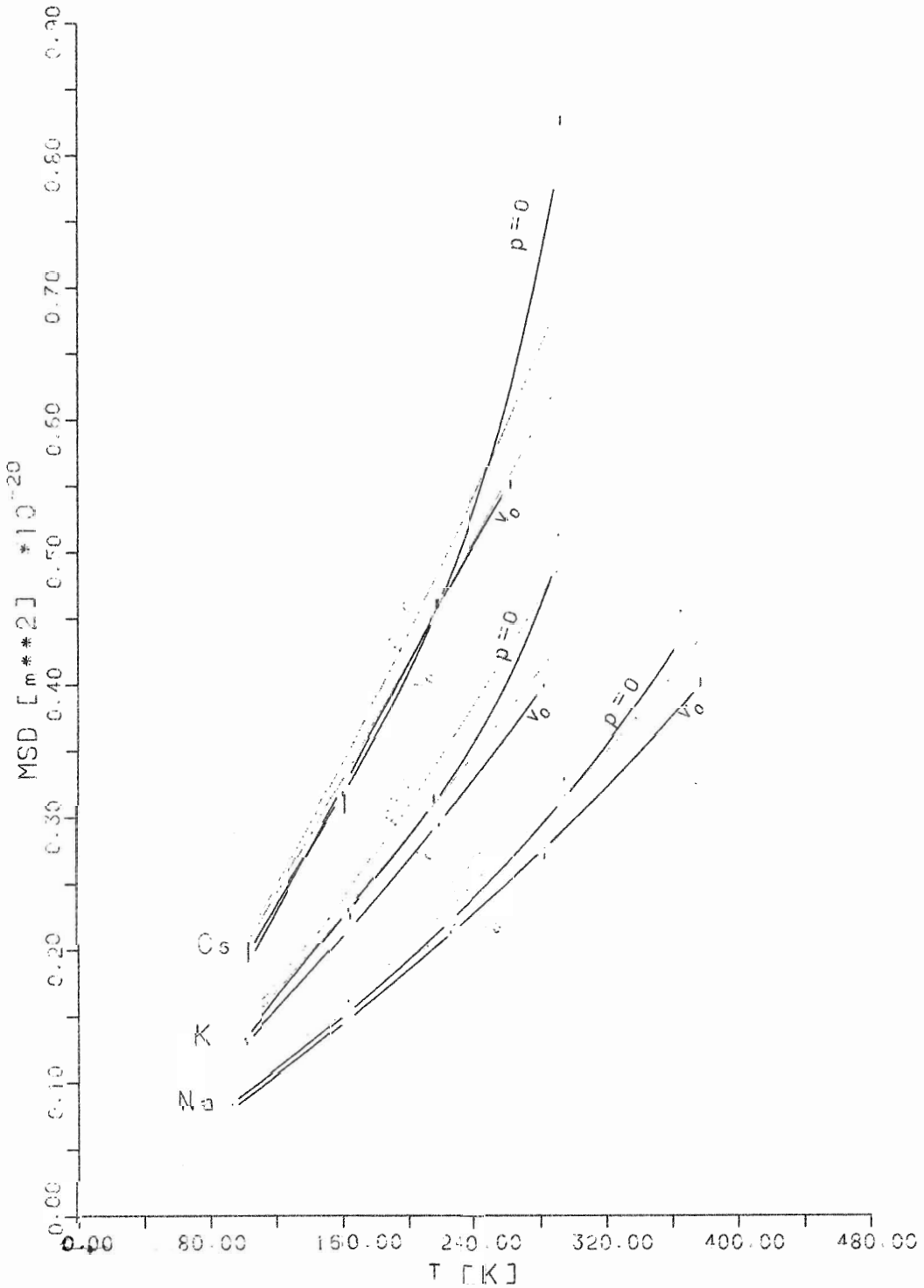

Fig. 4.1: MD and LD results for Na, K and Cs.
Red curves represent MD, green curves LD values.

4.4 Results for Rare Gas Solids

The converged tensors for the LD calculations for rare gas solids are given in Table 4.4. The scaling factor in this case is $2B(r_1)$ defined as

$$B(r_1) = \Phi''(r_1) + \frac{1}{r_1}\Phi'(r_1) \qquad (4.5)$$

The harmonic and anharmonic contributions to the LD value of MSD are presented in Table 4.5, where S_{QH}, S_1 and S_2 were defined in Eqs. (4.2), (4.3) and (4.4), the cubic contributions have again been provided by Shukla (1984). Table 4.6 finally contains the MD results and compares them with LD. The MSD as a function of temperature is graphed in Fig. 4.2. For this graph the MD values were approximated by smooth curves using a polynomial fit. To account for the fact that the size of the error bars differ largely from point to point, the MD values were weighted with the size of their error bars when determining the fitting function, thus giving points with small errors more importance. The fits were then used to obtain the MSD for a fixed temperature as a function of the a_1-ratio. This function is graphed for several temperatures in Fig. 4.3.

As discussed already in Sect. 4.1, the differences between MD and LD at lower temperatures are much larger than in the case of the alkali metals. Neglecting the fact that the MD curves are generally too low due to the influence of the small number of particles and concentrating only on the shape of the curves, it can be seen that the MD and LD curves look very similar for negative or zero a_1-ratio. This can be interpreted as a success of the $O(\lambda^2)$ perturbation theory in these cases. For positive values of a_1 the two curves start to exhibit an opposite behaviour. Starting with $a_1=0.02$ at high temperatures and for the higher ratios at progressively lower temperatures, the curves bend in opposite directions. This manifests a breakdown of the $O(\lambda^2)$-perturbation theory. The T-dependence of the λ^2 LD results is $c_1 T + c_2 T^2$. This is insufficient to represent the MD results. Addition of a term $c_3 T^3$ allows to adequately fit the points obtained from MD, except for the highest ratio $a_1 = 0.10$, where a term $c_4 T^4$ was required.

The most striking feature of the graph of MSD as a function of the a_1-ratio is that the MD curves are almost straight lines. This is quite a surprising relationship between the MSD and the potential parameter a_1.

Table 4.4: Converged S- and T-tensors used for the LD calculations for the rare gas solids

a_1	$S_{xx}(0)$	xx	xy	zz
0.10	0.5882	0.416647	-0.044647	0.446714
		0.2757	-0.0672	0.3040
0.08	0.6247	0.441013	-0.049501	0.474070
		0.3139	-0.0796	0.3463
0.06	0.6665	0.468700	-0.055258	0.505230
		0.3614	-0.0956	0.3987
0.04	0.7150	0.500492	-0.062175	0.541091
		0.4216	-0.1165	0.4646
0.02	0.7719	0.537461	-0.070617	0.582873
		0.4997	-0.1448	0.5496
0.00	0.8400	0.581115	-0.081115	0.632283
		0.6041	-0.1842	0.6622
-0.02	0.9233	0.633663	-0.094480	0.691797
		0.7494	-0.2416	0.8170
-0.04	1.0281	0.698507	-0.112019	0.765184
		0.9627	-0.3302	1.0401
-0.06	1.1655	0.781255	-0.135992	0.858552
		1.3003	-0.4787	1.3846
-0.08	1.3572	0.892105	-0.170749	0.982733
		1.903272	-0.7623	1.9770
-0.10	1.6557	1.052765	-0.226220	1.159924
		3.2602	-1.4577	3.2301

The last three numbers in the first row of every ratio are the xx, xy and zz components of the tensor $S_{\alpha\beta}(0)-S_{\alpha\beta}(1)$, the second row gives the same component of the tensor $T_{\alpha\beta}(0)-T_{\alpha\beta}(1)$. The yy, yx components are equal to the xx, xy components respectively, all other components are zero.

Table 4.5: LD results for MSD in rare gas solids

a_1	S_{QH}	S_1	S_2
0.10	0.018125	-0.0470450	0.0372167
0.08	0.016317	-0.0368479	0.0290764
0.06	0.014345	-0.0274503	0.0216367
0.04	0.012194	-0.0190704	0.0150472
0.02	0.009857	-0.0119575	0.0094761
0.00	0.007350	-0.0063750	0.0051008
-0.02	0.004736	-0.0025507	0.0024190
-0.04	0.002215	-0.0005506	0.0004635

Table 4.6: MD results for MSD in rare gas solids and comparison with LD

a_1	$k_B T/\epsilon$	QH	AH	QH+AH	MD
0.10	0.647(23)	35.156	-12.325	22.831	34.15(50)
	0.485(18)	26.383	-6.941	19.442	23.58(38)
	0.372(12)	20.200	-4.069	16.169	16.40(28)
	0.237(7)	12.858	-1.649	11.209	10.46(6)
	0.125(5)	6.802	-0.461	6.338	5.75(2)
0.08	0.657(22)	32.150	-10.057	22.057	31.81(34)
	0.472(17)	23.126	-5.203	17.922	19.05(25)
	0.362(13)	17.719	-3.055	14.664	14.62(5)
	0.244(8)	11.946	-1.389	10.558	9.74(4)
	0.121(5)	5.944	-0.344	5.600	5.08(2)
0.06	0.700(23)	30.116	-8.542	21.574	26.49(36)
	0.499(18)	21.482	-4.346	17.136	18.22(26)
	0.371(15)	15.977	-2.404	13.573	12.84(5)
	0.242(9)	10.419	-1.022	9.397	8.58(12)
	0.119(5)	5.127	-0.248	4.879	4.30(2)
0.04	0.748(28)	27.356	-6.750	20.606	23.37(20)
	0.531(19)	19.438	-3.408	16.030	16.15(14)
	0.373(12)	13.660	-1.683	11.977	11.11(5)
	0.246(8)	9.011	-0.732	8.279	7.40(3)
	0.122(5)	4.437	-0.178	4.260	3.70(2)
0.02	0.603(20)	17.875	-2.720	15.155	14.50(10)
	0.488(17)	14.416	-1.769	12.647	11.73(4)
	0.368(13)	10.872	-1.006	9.866	8.80(3)
	0.243(9)	7.181	-0.439	6.742	5.73(2)
	0.122(7)	3.605	-0.111	3.494	2.94(1)
0.00	0.680(25)	15.005	-1.770	13.233	11.84(6)
	0.486(19)	10.720	-0.903	9.816	8.50(4)
	0.370(12)	8.157	-0.523	7.634	6.68(3)
	0.245(9)	5.397	-0.229	5.168	4.40(2)
	0.120(5)	2.651	-0.055	2.595	2.24(1)
-0.02	0.610(19)	8.665	-0.147	8.518	6.90(5)
	0.473(15)	6.717	-0.088	6.628	5.10(3)
	0.367(11)	5.208	-0.053	5.155	4.20(2)
	0.245(7)	3.484	-0.024	3.461	2.77(2)
	0.118(5)	1.677	-0.006	1.672	1.39(1)
-0.04	0.591(19)	3.930	-0.091	3.839	3.14(2)
	0.469(14)	2.465	-0.057	3.059	2.47(2)
	0.366(11)	1.957	-0.035	2.396	1.96(1)
	0.242(9)	1.610	-0.015	1.594	1.30(1)
	0.121(7)	0.805	-0.004	0.801	0.67(1)

MSD values are multiplied by 1000, values in parentheses give the error of the preceding value in units of the last digit.

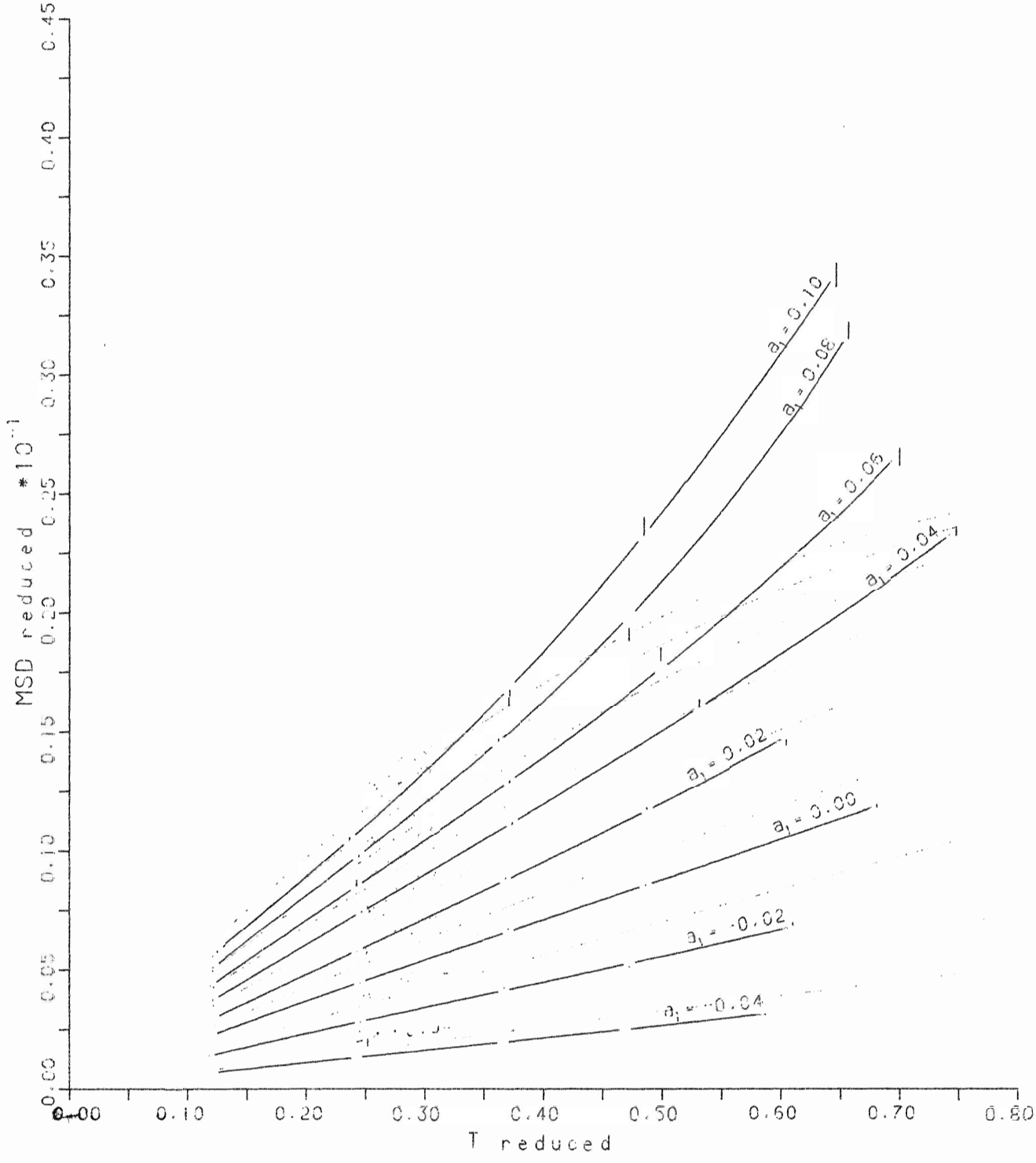

Fig. 4.2: MD and LD results for rare gas solids as a function of the reduced temperature.
Red curves represent MD, green curves LD values.

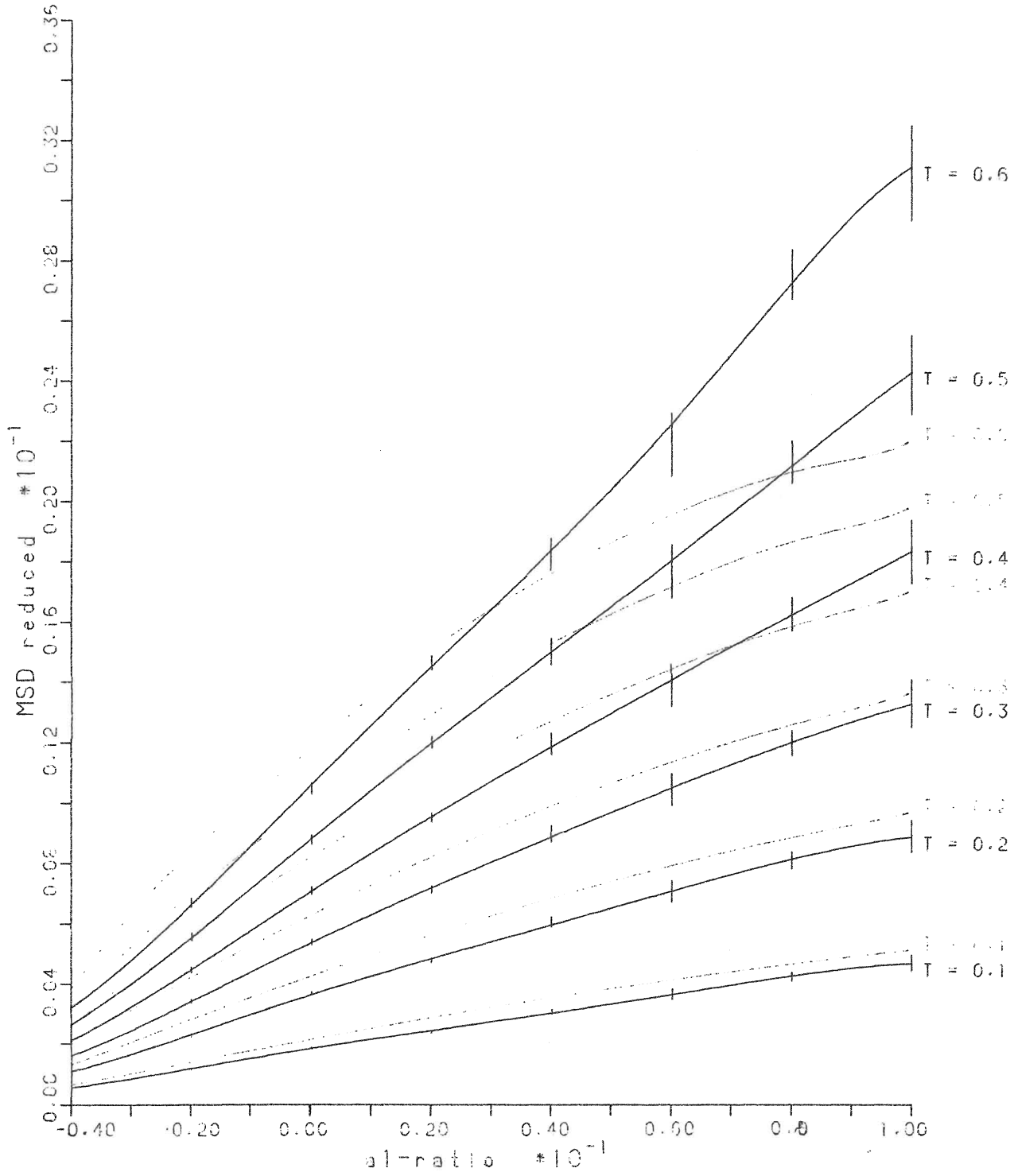

Fig. 4.3: MD and LD results for rare gas solids as a function of the a_1-ratio.
Red curves represent MD, green curves LD values.

CHAPTER 5

CONCLUSIONS

The objective of this thesis was to perform the calculation of the mean square displacement in alkali metals and rare gas solids by the molecular dynamics and lattice dynamics methods. As shown in the preceding chapter, this has been accomplished.

In the case of the alkalis, the agreement between the results from the two methods was excellent for all temperatures in all cases except one (zero pressure results for Cs at $T \sim T_m$). We conclude from this the adequacy of the $O(\lambda^2)$ perturbation theory for this group of metals.

From the calculations done for the rare gas solids we conclude that the $O(\lambda^2)$ perturbation theory works well for high densities, while at lower densities higher (eg. λ^4) order contributions become important. We also conclude from our data, that the simulations of rare gas solids using 256 particles are not converged with respect to the size of the ensemble.

Multa tulit fecitque puer, sudavit et alsit

Horatius

REFERENCES

B. J. Alder and T. E. Wainwright, J. Chem. Phys. **31** (1959) 459

N. W. Ashcroft, Phys. Letters 23 (1966) 48

M. Born and K. Huang, "Dynamical Theory of Crystal Lattices" (Oxford Univ. Press, Amen House, London, England, 1954), 140

V. V. Goldman, Phys. Rev, **174** (1968) 1041

J. L. Lebowitz, J. K. Percus and L. Verlet, Phys. Rev. **153** (1967) 250

A. A. Maradudin and P. A. Flinn, Phys. Rev. **129** (1963) 2529

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, J. Chem. Phys. **21** (1953) 1087

D. Pines, "Elementary Excitations in Solids" (Benjamin, New York, 1964), 34

D. L. Price, K. S. Singwi and M. P. Tosi, Phys. Rev. B **2** (1970) 2983

A. Rahman, Phys. Rev. 136 (1964) A405

M. J. L. Sangster and M. Dixon, Adv. Phys. **25** (1976) 296

R. C. Shukla, Int. J. Thermophys. **1** (1980) 73

R. C. Shukla, Phys. Rev. B **23** (1981) 3087

R. C. Shukla, unpublished work (1984)

R. C. Shukla and R. A. MacDonald, High Temp.-High Pressures **12** (1980) 291

R. C. Shukla and R. D. Mountain, Phys. Rev. B **25** (1982) 3649

REFERENCES

R. C. Shukla and C. A. Plint in "Proceedings of the Eighth Symposium on Thermophysical Properties", 2: "Thermophysical Properties of Solids and of Selected Fluids for Energy Technology", edited by Jan V. Sengers (Am. Soc. Mech. Eng., New York, 1982), 77

R. C. Shukla and R. Taylor, Phys. Rev. B **9** (1974) 4116

R. C. Shukla and L. Wilk, Phys. Rev. B **10** (1974) 3660

L. Van Hove, "Quantum Theory of Many Particle Systems" (Benjamin, New York, 1961)

P. Vashishta and K. S. Singwi, Phys. Rev. B **6** (1972) 875

L. Verlet, Phys. Rev. **159** (1967) 98

APPENDIX I

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

This appendix contains the tables of the converged $S_{\alpha\beta}$-and $T_{\alpha\beta}$-tensors used in determining the quasiharmonic and quartic contributions to the MSD in Na, K and Cs. The tables are given in the order of increasing lattice constant.

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.1: Converged S- and T-tensors for Na, a = 4.225 Å

alphal = 1.18770 N/m, Sxx(0) = 0.26013

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.175436	-0.035098	-0.035098)	(0.09121	-0.03460	-0.03460)
(-0.035098	0.175436	-0.035098)	(-0.03460	0.09121	-0.03460)
(-0.035098	-0.035098	0.175436)	(-0.03460	-0.03460	0.09121)
Shell 2					
(0.226073	0.000000	0.000000)	(0.15278	0.00000	0.00000)
(0.000000	0.188707	0.000000)	(0.00000	0.09881	0.00000)
(0.000000	0.000000	0.188707)	(0.00000	0.00000	0.09881)
Shell 3					
(0.214008	-0.019633	0.000000)	(0.16090	-0.05735	0.00000)
(-0.019633	0.214008	0.000000)	(-0.05735	0.16090	0.00000)
(0.000000	0.000000	0.230016)	(0.00000	0.00000	0.16267)
Shell 4					
(0.232493	-0.006231	-0.006231)	(0.21661	-0.03124	-0.03124)
(-0.006231	0.219300	-0.011706)	(-0.03124	0.17075	-0.02374)
(-0.006231	-0.011706	0.219300)	(-0.03124	-0.02374	0.17075)
Shell 5					
(0.207082	-0.023482	-0.023482)	(0.17265	-0.05784	-0.05784)
(-0.023482	0.207082	-0.023482)	(-0.05784	0.17265	-0.05784)
(-0.023482	-0.023482	0.207082)	(-0.05784	-0.05784	0.17265)
Shell 6					
(0.242552	0.000000	0.000000)	(0.27650	0.00000	0.00000)
(0.000000	0.220415	0.000000)	(0.00000	0.19043	0.00000)
(0.000000	0.000000	0.220415)	(0.00000	0.00000	0.19043)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.2: Converged S- and T-tensors for Na, a = 4.234 Å

alphal = 1.16994 N/m, Sxx(0) = 0.26065

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.175777	-0.035203	-0.035203)	(0.09184	-0.03489	-0.03489)
(-0.035203	0.175777	-0.035203)	(-0.03489	0.09184	-0.03489)
(-0.035203	-0.035203	0.175777)	(-0.03489	-0.03489	0.09184)
Shell 2					
(0.226888	0.000000	0.000000)	(0.15412	0.00000	0.00000)
(0.000000	0.188993	0.000000)	(0.00000	0.09939	0.00000)
(0.000000	0.000000	0.188993)	(0.00000	0.00000	0.09939)
Shell 3					
(0.214473	-0.019758	0.000000)	(0.16204	-0.05783	0.00000)
(-0.019758	0.214473	0.000000)	(-0.05783	0.16204	0.00000)
(0.000000	0.000000	0.230499)	(0.00000	0.00000	0.16375)
Shell 4					
(0.233128	-0.006217	-0.006217)	(0.21833	-0.03144	-0.03144)
(-0.006217	0.219718	-0.011795)	(-0.03144	0.17180	-0.02399)
(-0.006217	-0.011795	0.219718)	(-0.03144	-0.02399	0.17180)
Shell 5					
(0.207455	-0.023555	-0.023555)	(0.17372	-0.05829	-0.05829)
(-0.023555	0.207455	-0.023555)	(-0.05829	0.17372	-0.05829)
(-0.023555	-0.023555	0.207455)	(-0.05829	-0.05829	0.17372)
Shell 6					
(0.243145	0.000000	0.000000)	(0.27866	0.00000	0.00000)
(0.000000	0.220763	0.000000)	(0.00000	0.19143	0.00000)
(0.000000	0.000000	0.220763)	(0.00000	0.00000	0.19143)

Table I.3: Converged S- and T-tensors for Na, a = 4.251 Å

alpha1 = 1.13698 N/m, Sxx(0) = 0.26167

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.176440	-0.035420	-0.035420)	(0.09310	-0.03548	-0.03548)
	(-0.035420	0.176440	-0.035420)	(-0.03548	0.09310	-0.03548)
	(-0.035420	-0.035420	0.176440)	(-0.03548	-0.03548	0.09310)
Shell 2						
	(0.228458	0.000000	0.000000)	(0.15680	0.00000	0.00000)
	(0.000000	0.189561	0.000000)	(0.00000	0.10057	0.00000)
	(0.000000	0.000000	0.189561)	(0.00000	0.00000	0.10057)
Shell 3						
	(0.215373	-0.020010	0.000000)	(0.16435	-0.05882	0.00000)
	(-0.020010	0.215373	0.000000)	(-0.05882	0.16435	0.00000)
	(0.000000	0.000000	0.231459)	(0.00000	0.00000	0.16597)
Shell 4						
	(0.234359	-0.006190	-0.006190)	(0.22178	-0.03184	-0.03184)
	(-0.006190	0.220541	-0.011972)	(-0.03184	0.17395	-0.02449)
	(-0.006190	-0.011972	0.220541)	(-0.03184	-0.02449	0.17395)
Shell 5						
	(0.208171	-0.023708	-0.023708)	(0.17588	-0.05921	-0.05921)
	(-0.023708	0.208171	-0.023708)	(-0.05921	0.17588	-0.05921)
	(-0.023708	-0.023708	0.208171)	(-0.05921	-0.05921	0.17588)
Shell 6						
	(0.244297	0.000000	0.000000)	(0.28300	0.00000	0.00000)
	(0.000000	0.221446	0.000000)	(0.00000	0.19349	0.00000)
	(0.000000	0.000000	0.221446)	(0.00000	0.00000	0.19349)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.4: Converged S- and T-tensors for Na, a = 4.262 Å

alphal = 1.11604 N/m, Sxx(0) = 0.26238

S (0) - S (Shell)	T (0) - T (Shell)
Shell 1	
(0.176896 -0.035569 -0.035569)	(0.09398 -0.03589 -0.03589)
(-0.035569 0.176896 -0.035569)	(-0.03589 0.09398 -0.03589)
(-0.035569 -0.035569 0.176896)	(-0.03589 -0.03589 0.09398)
Shell 2	
(0.229526 0.000000 0.000000)	(0.15866 0.00000 0.00000)
(0.000000 0.189956 0.000000)	(0.00000 0.10140 0.00000)
(0.000000 0.000000 0.189956)	(0.00000 0.00000 0.10140)
Shell 3	
(0.215993 -0.020181 0.000000)	(0.16596 -0.05950 0.00000)
(-0.020181 0.215993 0.000000)	(-0.05950 0.16596 0.00000)
(0.000000 0.000000 0.232122)	(0.00000 0.00000 0.16752)
Shell 4	
(0.235204 -0.006172 -0.006172)	(0.22418 -0.03213 -0.03213)
(-0.006172 0.221110 -0.012092)	(-0.03213 0.17544 -0.02484)
(-0.006172 -0.012092 0.221110)	(-0.03213 -0.02484 0.17544)
Shell 5	
(0.208666 -0.023814 -0.023814)	(0.17739 -0.05985 -0.05985)
(-0.023814 0.208666 -0.023814)	(-0.05985 0.17739 -0.05985)
(-0.023814 -0.023814 0.208666)	(-0.05985 -0.05985 0.17739)
Shell 6	
(0.245088 0.000000 0.000000)	(0.28601 0.00000 0.00000)
(0.000000 0.221920 0.000000)	(0.00000 0.19493 0.00000)
(0.000000 0.000000 0.221920)	(0.00000 0.00000 0.19493)

Table I.5: Converged S- and T-tensors for Na, a = 4.288 Å

alpha1 = 1.06775 N/m, Sxx(0) = 0.26422

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.178069	-0.035957	-0.035957)	(0.09627	-0.03695	-0.03695)
(-0.035957	0.178069	-0.035957)	(-0.03695	0.09627	-0.03695)
(-0.035957	-0.035957	0.178069)	(-0.03695	-0.03695	0.09627)
Shell 2					
(0.232231	0.000000	0.000000)	(0.16348	0.00000	0.00000)
(0.000000	0.190985	0.000000)	(0.00000	0.10359	0.00000)
(0.000000	0.000000	0.190985)	(0.00000	0.00000	0.10359)
Shell 3					
(0.217592	-0.020615	0.000000)	(0.17018	-0.06129	0.00000)
(-0.020615	0.217592	0.000000)	(-0.06129	0.17018	0.00000)
(0.000000	0.000000	0.233841)	(0.00000	0.00000	0.17158)
Shell 4					
(0.237369	-0.006131	-0.006131)	(0.23043	-0.03287	-0.03287)
(-0.006131	0.222585	-0.012400)	(-0.03287	0.17939	-0.02575)
(-0.006131	-0.012400	0.222585)	(-0.03287	-0.02575	0.17939)
Shell 5					
(0.209948	-0.024092	-0.024092)	(0.18136	-0.06153	-0.06153)
(-0.024092	0.209948	-0.024092)	(-0.06153	0.18136	-0.06153)
(-0.024092	-0.024092	0.209948)	(-0.06153	-0.06153	0.18136)
Shell 6					
(0.247119	0.000000	0.000000)	(0.29387	0.00000	0.00000)
(0.000000	0.223156	0.000000)	(0.00000	0.19874	0.00000)
(0.000000	0.000000	0.223156)	(0.00000	0.00000	0.19874)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.6: Converged S- and T-tensors for Na, a = 4.309 Å

alpha1 = 1.02996 N/m, Sxx(0) = 0.26588

S (0) - S (Shell)	T (0) - T (Shell)
Shell 1	
(0.179121 -0.036307 -0.036307)	(0.09838 -0.03792 -0.03792)
(-0.036307 0.179121 -0.036307)	(-0.03792 0.09838 -0.03792)
(-0.036307 -0.036307 0.179121)	(-0.03792 -0.03792 0.09838)
Shell 2	
(0.234618 0.000000 0.000000)	(0.16787 0.00000 0.00000)
(0.000000 0.191921 0.000000)	(0.00000 0.10560 0.00000)
(0.000000 0.000000 0.191921)	(0.00000 0.00000 0.10560)
Shell 3	
(0.219032 -0.020999 0.000000)	(0.17404 -0.06292 0.00000)
(-0.020999 0.219032 0.000000)	(-0.06292 0.17404 0.00000)
(0.000000 0.000000 0.235393)	(0.00000 0.00000 0.17533)
Shell 4	
(0.239303 -0.006098 -0.006098)	(0.23615 -0.03355 -0.03355)
(-0.006098 0.223918 -0.012673)	(-0.03355 0.18303 -0.02658)
(-0.006098 -0.012673 0.223918)	(-0.03355 -0.02658 0.18303)
Shell 5	
(0.211104 -0.024346 -0.024346)	(0.18501 -0.06306 -0.06306)
(-0.024346 0.211104 -0.024346)	(-0.06306 0.18501 -0.06306)
(-0.024346 -0.024346 0.211104)	(-0.06306 -0.06306 0.18501)
Shell 6	
(0.248935 0.000000 0.000000)	(0.30105 0.00000 0.00000)
(0.000000 0.224279 0.000000)	(0.00000 0.20228 0.00000)
(0.000000 0.000000 0.224279)	(0.00000 0.00000 0.20228)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.7: Converged S- and T-tensors for K, a = 5.225 Å

alphal = 0.76797 N/m, Sxx(0) = 0.24730

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.175436	-0.035098	-0.035098)	(0.09121	-0.03460	-0.03460)
	(-0.035098	0.175436	-0.035098)	(-0.03460	0.09121	-0.03460)
	(-0.035098	-0.035098	0.175436)	(-0.03460	-0.03460	0.09121)
Shell 2						
	(0.226073	0.000000	0.000000)	(0.15278	0.00000	0.00000)
	(0.000000	0.188707	0.000000)	(0.00000	0.09881	0.00000)
	(0.000000	0.000000	0.188707)	(0.00000	0.00000	0.09881)
Shell 3						
	(0.214008	-0.019633	0.000000)	(0.16090	-0.05735	0.00000)
	(-0.019633	0.214008	0.000000)	(-0.05735	0.16090	0.00000)
	(0.000000	0.000000	0.230016)	(0.00000	0.00000	0.16267)
Shell 4						
	(0.232493	-0.006231	-0.006231)	(0.21661	-0.03124	-0.03124)
	(-0.006231	0.219300	-0.011706)	(-0.03124	0.17075	-0.02374)
	(-0.006231	-0.011706	0.219300)	(-0.03124	-0.02374	0.17075)
Shell 5						
	(0.207082	-0.023482	-0.023482)	(0.17265	-0.05784	-0.05784)
	(-0.023482	0.207082	-0.023482)	(-0.05784	0.17265	-0.05784)
	(-0.023482	-0.023482	0.207082)	(-0.05784	-0.05784	0.17265)
Shell 6						
	(0.242552	0.000000	0.000000)	(0.27650	0.00000	0.00000)
	(0.000000	0.220415	0.000000)	(0.00000	0.19043	0.00000)
	(0.000000	0.000000	0.220415)	(0.00000	0.00000	0.19043)

Table I.8: Converged S- and T-tensors for K, a = 5.261 Å

alphal = 0.73292 N/m, Sxx(0) = 0.24778

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.168460	-0.032347	-0.032347)	(0.07738	-0.02827	-0.02827)
	(-0.032347	0.168460	-0.032347)	(-0.02827	0.07738	-0.02827)
	(-0.032347	-0.032347	0.168460)	(-0.02827	-0.02827	0.07738)
Shell 2						
	(0.212132	0.000000	0.000000)	(0.12563	0.00000	0.00000)
	(0.000000	0.181534	0.000000)	(0.00000	0.08470	0.00000)
	(0.000000	0.000000	0.181534)	(0.00000	0.00000	0.08470)
Shell 3						
	(0.204481	-0.016894	0.000000)	(0.13484	-0.04577	0.00000)
	(-0.016894	0.204481	0.000000)	(-0.04577	0.13484	0.00000)
	(0.000000	0.000000	0.218683)	(0.00000	0.00000	0.13694)
Shell 4						
	(0.219797	-0.006151	-0.006151)	(0.17847	-0.02565	-0.02565)
	(-0.006151	0.209596	-0.010022)	(-0.02565	0.14457	-0.01852)
	(-0.006151	-0.010022	0.209596)	(-0.02565	-0.01852	0.14457)
Shell 5						
	(0.198968	-0.021303	-0.021303)	(0.14619	-0.04671	-0.04671)
	(-0.021303	0.198968	-0.021303)	(-0.04671	0.14619	-0.04671)
	(-0.021303	-0.021303	0.198968)	(-0.04671	-0.04671	0.14619)
Shell 6						
	(0.229728	0.000000	0.000000)	(0.22700	0.00000	0.00000)
	(0.000000	0.211744	0.000000)	(0.00000	0.16297	0.00000)
	(0.000000	0.000000	0.211744)	(0.00000	0.00000	0.16297)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.9: Converged S- and T-tensors for K, a = 5.277 Å

alphal = 0.71777 N/m, Sxx(0) = 0.24800

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.168651	-0.032399	-0.032399)	(0.07767	-0.02842	-0.02842)
	(-0.032399	0.168651	-0.032399)	(-0.02842	0.07767	-0.02842)
	(-0.032399	-0.032399	0.168651)	(-0.02842	-0.02842	0.07767)
Shell 2						
	(0.212770	0.000000	0.000000)	(0.12639	0.00000	0.00000)
	(0.000000	0.181641	0.000000)	(0.00000	0.08492	0.00000)
	(0.000000	0.000000	0.181641)	(0.00000	0.00000	0.08492)
Shell 3						
	(0.204727	-0.016993	0.000000)	(0.13535	-0.04600	0.00000)
	(-0.016993	0.204727	0.000000)	(-0.04600	0.13535	0.00000)
	(0.000000	0.000000	0.218916)	(0.00000	0.00000	0.13738)
Shell 4						
	(0.220192	-0.006122	-0.006122)	(0.17931	-0.02572	-0.02572)
	(-0.006122	0.209786	-0.010088)	(-0.02572	0.14496	-0.01866)
	(-0.006122	-0.010088	0.209786)	(-0.02572	-0.01866	0.14496)
Shell 5						
	(0.199140	-0.021327	-0.021327)	(0.14659	-0.04690	-0.04690)
	(-0.021327	0.199140	-0.021327)	(-0.04690	0.14659	-0.04690)
	(-0.021327	-0.021327	0.199140)	(-0.04690	-0.04690	0.14659)
Shell 6						
	(0.230068	0.000000	0.000000)	(0.22802	0.00000	0.00000)
	(0.000000	0.211867	0.000000)	(0.00000	0.16324	0.00000)
	(0.000000	0.000000	0.211867)	(0.00000	0.00000	0.16324)

Table I.10: Converged S- and T-tensors for K, a = 5.305 Å

alpha1 = 0.69184 N/m, Sxx(0) = 0.24846

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.169028	-0.032504	-0.032504)	(0.07825	-0.02871	-0.02871)
	(-0.032504	0.169028	-0.032504)	(-0.02871	0.07825	-0.02871)
	(-0.032504	-0.032504	0.169028)	(-0.02871	-0.02871	0.07825)
Shell 2						
	(0.213967	0.000000	0.000000)	(0.12785	0.00000	0.00000)
	(0.000000	0.181871	0.000000)	(0.00000	0.08538	0.00000)
	(0.000000	0.000000	0.181871)	(0.00000	0.00000	0.08538)
Shell 3						
	(0.205218	-0.017178	0.000000)	(0.13639	-0.04647	0.00000)
	(-0.017178	0.205218	0.000000)	(-0.04647	0.13639	0.00000)
	(0.000000	0.000000	0.219388)	(0.00000	0.00000	0.13829)
Shell 4						
	(0.220957	-0.006073	-0.006073)	(0.18097	-0.02585	-0.02585)
	(-0.006073	0.210178	-0.010211)	(-0.02585	0.14576	-0.01892)
	(-0.006073	-0.010211	0.210178)	(-0.02585	-0.01892	0.14576)
Shell 5						
	(0.199494	-0.021379	-0.021379)	(0.14742	-0.04729	-0.04729)
	(-0.021379	0.199494	-0.021379)	(-0.04729	0.14742	-0.04729)
	(-0.021379	-0.021379	0.199494)	(-0.04729	-0.04729	0.14742)
Shell 6						
	(0.230734	0.000000	0.000000)	(0.23004	0.00000	0.00000)
	(0.000000	0.212136	0.000000)	(0.00000	0.16384	0.00000)
	(0.000000	0.000000	0.212136)	(0.00000	0.00000	0.16384)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.11: Converged S- and T-tensors for K, a = 5.343 Å

alpha1 = 0.65786 N/m, Sxx(0) = 0.24922

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.169623	-0.032673	-0.032673)	(0.07919	-0.02917	-0.02917)
(-0.032673	0.169623	-0.032673)	(-0.02917	0.07919	-0.02917)
(-0.032673	-0.032673	0.169623)	(-0.02917	-0.02917	0.07919)
Shell 2					
(0.215746	0.000000	0.000000)	(0.13011	0.00000	0.00000)
(0.000000	0.182267	0.000000)	(0.00000	0.08615	0.00000)
(0.000000	0.000000	0.182267)	(0.00000	0.00000	0.08615)
Shell 3					
(0.205999	-0.017450	0.000000)	(0.13805	-0.04719	0.00000)
(-0.017450	0.205999	0.000000)	(-0.04719	0.13805	0.00000)
(0.000000	0.000000	0.220152)	(0.00000	0.00000	0.13977)
Shell 4					
(0.222135	-0.006007	-0.006007)	(0.18359	-0.02608	-0.02608)
(-0.006007	0.210822	-0.010394)	(-0.02608	0.14710	-0.01932)
(-0.006007	-0.010394	0.210822)	(-0.02608	-0.01932	0.14710)
Shell 5					
(0.200074	-0.021472	-0.021472)	(0.14879	-0.04792	-0.04792)
(-0.021472	0.200074	-0.021472)	(-0.04792	0.14879	-0.04792)
(-0.021472	-0.021472	0.200074)	(-0.04792	-0.04792	0.14879)
Shell 6					
(0.231770	0.000000	0.000000)	(0.23324	0.00000	0.00000)
(0.000000	0.212604	0.000000)	(0.00000	0.16492	0.00000)
(0.000000	0.000000	0.212604)	(0.00000	0.00000	0.16492)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.12: Converged S- and T-tensors for Cs, a = 6.045 Å

alphal = 0.52402 N/m, Sxx(0) = 0.24442

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.166459	-0.031577	-0.031577)	(0.07378	-0.02659	-0.02659)
(-0.031577	0.166459	-0.031577)	(-0.02659	0.07378	-0.02659)
(-0.031577	-0.031577	0.166459)	(-0.02659	-0.02659	0.07378)
Shell 2					
(0.207617	0.000000	0.000000)	(0.11822	0.00000	0.00000)
(0.000000	0.179616	0.000000)	(0.00000	0.08115	0.00000)
(0.000000	0.000000	0.179616)	(0.00000	0.00000	0.08115)
Shell 3					
(0.201792	-0.016014	0.000000)	(0.12809	-0.04271	0.00000)
(-0.016014	0.201792	0.000000)	(-0.04271	0.12809	0.00000)
(0.000000	0.000000	0.215541)	(0.00000	0.00000	0.13042)
Shell 4					
(0.216024	-0.006179	-0.006179)	(0.16840	-0.02425	-0.02425)
(-0.006179	0.206924	-0.009507)	(-0.02425	0.13801	-0.01713)
(-0.006179	-0.009507	0.206924)	(-0.02425	-0.01713	0.13801)
Shell 5					
(0.196714	-0.020727	-0.020727)	(0.13951	-0.04385	-0.04385)
(-0.020727	0.196714	-0.020727)	(-0.04385	0.13951	-0.04385)
(-0.020727	-0.020727	0.196714)	(-0.04385	-0.04385	0.13951)
Shell 6					
(0.225977	0.000000	0.000000)	(0.21396	0.00000	0.00000)
(0.000000	0.209429	0.000000)	(0.00000	0.15627	0.00000)
(0.000000	0.000000	0.209429)	(0.00000	0.00000	0.15627)

Table I.13: Converged S- and T-tensors for Cs, a = 6.069 Å

alpha1 = 0.51015 N/m, Sxx(0) = 0.24441

S (0) - S (Shell)	T (0) - T (Shell)
Shell 1	
(0.166544 -0.031601 -0.031601)	(0.07392 -0.02668 -0.02668)
(-0.031601 0.166544 -0.031601)	(-0.02668 0.07392 -0.02668)
(-0.031601 -0.031601 0.166544)	(-0.02668 -0.02668 0.07392)
Shell 2	
(0.208187 0.000000 0.000000)	(0.11877 0.00000 0.00000)
(0.000000 0.179588 0.000000)	(0.00000 0.08120 0.00000)
(0.000000 0.000000 0.179588)	(0.00000 0.00000 0.08120)
Shell 3	
(0.201881 -0.016113 0.000000)	(0.12831 -0.04285 0.00000)
(-0.016113 0.201881 0.000000)	(-0.04285 0.12831 0.00000)
(0.000000 0.000000 0.215609)	(0.00000 0.00000 0.13056)
Shell 4	
(0.216272 -0.006134 -0.006134)	(0.16886 -0.02424 -0.02424)
(-0.006134 0.206950 -0.009568)	(-0.02424 0.13806 -0.01723)
(-0.006134 -0.009568 0.206950)	(-0.02424 -0.01723 0.13806)
Shell 5	
(0.196730 -0.020721 -0.020721)	(0.13955 -0.04393 -0.04393)
(-0.020721 0.196730 -0.020721)	(-0.04393 0.13955 -0.04393)
(-0.020721 -0.020721 0.196730)	(-0.04393 -0.04393 0.13955)
Shell 6	
(0.226155 0.000000 0.000000)	(0.21448 0.00000 0.00000)
(0.000000 0.209373 0.000000)	(0.00000 0.15612 0.00000)
(0.000000 0.000000 0.209373)	(0.00000 0.00000 0.15612)

Table I.14: Converged S- and T-tensors for Cs, a = 6.092 Å

alphal = 0.49714 N/m, Sxx(0) = 0.24453

S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1					
(0.166697	-0.031637	-0.031637)	(0.07414	-0.02681	-0.02681)
(-0.031637	0.166697	-0.031637)	(-0.02681	0.07414	-0.02681)
(-0.031637	-0.031637	0.166697)	(-0.02681	-0.02681	0.07414)
Shell 2					
(0.208840	0.000000	0.000000)	(0.11944	0.00000	0.00000)
(0.000000	0.179631	0.000000)	(0.00000	0.08132	0.00000)
(0.000000	0.000000	0.179631)	(0.00000	0.00000	0.08132)
Shell 3					
(0.202069	-0.016215	0.000000)	(0.12868	-0.04303	0.00000)
(-0.016215	0.202069	0.000000)	(-0.04303	0.12868	0.00000)
(0.000000	0.000000	0.215768)	(0.00000	0.00000	0.13084)
Shell 4					
(0.216622	-0.006095	-0.006095)	(0.16952	-0.02427	-0.02427)
(-0.006095	0.207070	-0.009634)	(-0.02427	0.13826	-0.01735)
(-0.006095	-0.009634	0.207070)	(-0.02427	-0.01735	0.13826)
Shell 5					
(0.196842	-0.020727	-0.020727)	(0.13977	-0.04406	-0.04406)
(-0.020727	0.196842	-0.020727)	(-0.04406	0.13977	-0.04406)
(-0.020727	-0.020727	0.196842)	(-0.04406	-0.04406	0.13977)
Shell 6					
(0.226438	0.000000	0.000000)	(0.21526	0.00000	0.00000)
(0.000000	0.209416	0.000000)	(0.00000	0.15616	0.00000)
(0.000000	0.000000	0.209416)	(0.00000	0.00000	0.15616)

Table I.15: Converged S- and T-tensors for Cs, a = 6.119 Å

alpha1 = 0.48221 N/m, Sxx(0) = 0.24472

	S (0) - S (Shell)			T (0) - T (Shell)		
Shell 1						
	(0.166901	-0.031688	-0.031688)	(0.07443	-0.02697	-0.02697)
	(-0.031688	0.166901	-0.031688)	(-0.02697	0.07443	-0.02697)
	(-0.031688	-0.031688	0.166901)	(-0.02697	-0.02697	0.07443)
Shell 2						
	(0.209652	0.000000	0.000000)	(0.12030	0.00000	0.00000)
	(0.000000	0.179707	0.000000)	(0.00000	0.08150	0.00000)
	(0.000000	0.000000	0.179707)	(0.00000	0.00000	0.08150)
Shell 3						
	(0.202325	-0.016340	0.000000)	(0.12918	-0.04327	0.00000)
	(-0.016340	0.202325	0.000000)	(-0.04327	0.12918	0.00000)
	(0.000000	0.000000	0.215990)	(0.00000	0.00000	0.13124)
Shell 4						
	(0.217074	-0.006051	-0.006051)	(0.17039	-0.02431	-0.02431)
	(-0.006051	0.207244	-0.009716)	(-0.02431	0.13856	-0.01750)
	(-0.006051	-0.009716	0.207244)	(-0.02431	-0.01750	0.13856)
Shell 5						
	(0.197002	-0.020741	-0.020741)	(0.14009	-0.04424	-0.04424)
	(-0.020741	0.197002	-0.020741)	(-0.04424	0.14009	-0.04424)
	(-0.020741	-0.020741	0.197002)	(-0.04424	-0.04424	0.14009)
Shell 6						
	(0.226808	0.000000	0.000000)	(0.21630	0.00000	0.00000)
	(0.000000	0.209497	0.000000)	(0.00000	0.15627	0.00000)
	(0.000000	0.000000	0.209497)	(0.00000	0.00000	0.15627)

TABLES OF S- AND T-TENSORS FOR NA, K AND CS

Table I.16: Converged S- and T-tensors for Cs, a = 6.163 Å

alphal = 0.45863 N/m, Sxx(0) = 0.24517

S (0) - S (Shell)	T (0) - T (Shell)
Shell 1	
(0.167320 -0.031797 -0.031797)	(0.07504 -0.02729 -0.02729)
(-0.031797 0.167320 -0.031797)	(-0.02729 0.07504 -0.02729)
(-0.031797 -0.031797 0.167320)	(-0.02729 -0.02729 0.07504)
Shell 2	
(0.211132 0.000000 0.000000)	(0.12195 0.00000 0.00000)
(0.000000 0.179920 0.000000)	(0.00000 0.08194 0.00000)
(0.000000 0.000000 0.179920)	(0.00000 0.00000 0.08194)
Shell 3	
(0.202860 -0.016568 0.000000)	(0.13026 -0.04376 0.00000)
(-0.016568 0.202860 0.000000)	(-0.04376 0.13026 0.00000)
(0.000000 0.000000 0.216483)	(0.00000 0.00000 0.13214)
Shell 4	
(0.217957 -0.005980 -0.005980)	(0.17217 -0.02443 -0.02443)
(-0.005980 0.207645 -0.009866)	(-0.02443 0.13931 -0.01779)
(-0.005980 -0.009866 0.207645)	(-0.02443 -0.01779 0.13931)
Shell 5	
(0.197369 -0.020786 -0.020786)	(0.14087 -0.04463 -0.04463)
(-0.020786 0.197369 -0.020786)	(-0.04463 0.14087 -0.04463)
(-0.020786 -0.020786 0.197369)	(-0.04463 -0.04463 0.14087)
Shell 6	
(0.227553 0.000000 0.000000)	(0.21844 0.00000 0.00000)
(0.000000 0.209740 0.000000)	(0.00000 0.15674 0.00000)
(0.000000 0.000000 0.209740)	(0.00000 0.00000 0.15674)

APPENDIX II

SOME DETAILS OF THE MD PROGRAM

This appendix describes some details of the MD program which are not essential for the understanding of the MD method or the computations done for this thesis, but are of interest for a potential user of MD. It also includes some details on verification of the program.

II.1 Program Features

The program was for the main part written in FORTRAN IV. However it was found that cpu usage can be reduced by 40 % on the B6700 if the procedure for computing the accelerations, which is the most time consuming part of the program, is coded in ALGOL.

II.1.1 Initialization Procedures

The program can initialize a cubic box of particles forming an fcc or bcc lattice and supply them with normally distributed random velocities corresponding to some predefined temperature. After this initialization the total linear and angular momentum is eliminated.

A table of the potential function can be generated (for Lennard-Jones potential) or read from a disk file.

II.1.2 CHECKPOINT/RESTART Procedures

An essential feature of the program is the ability to save its current status. This is done by writing all the program's data to a disk file (CHECKPOINT). In a later run of the program, this CHECKPOINT file can be read in to continue (RESTART) a simulation from the point at which the CHECKPOINT was done. This facility is important for several reasons:

- during the initial phase of reaching thermal equilibrium at a certain temperature, the simulation must be run for a while until the temperature is sufficiently stable. Then the velocities must be scaled to get closer to the desired temperature. This can generally not be done interactively because of the high time consumption. Storing all the relevant information on a disk file is

the only reasonable method to hand over the program's state from one batch job to the next.

- Even after equilibrium has been reached at the right temperature, the whole simulation cannot be done in a single batch job due to the limit of two hours cpu time per batch job on Brock's B6700. The simulation must therefore be broken up into several jobs, which is only possible with a CHECKPOINT/RESTART facility.

- The effort required to reach thermal equilibrium can be reduced dramatically if a new simulation does not have to start from scratch. The CHECKPOINT/RESTART feature together with internally representing all quantities in dimensionless units allowed for example to turn a simulation of Rb into one for K by just reading in a new potential table. The resulting K system was almost in equilibrium right from the beginning.

- An additional benefit of the CHECKPOINT/RESTART facility is that it allows to minimize losses of time due to computer system failures. To this end a CHECKPOINT is performed regularly during the simulation. If the system crashes, the simulation can be restarted from the last CHECKPOINT rather than from the begin of the job.

The program was set up as to perform a CHECKPOINT at regular intervals (approximately every 20 minutes of cpu time) and again before it terminates. It also inquires the maximum amount of cpu time it is allowed to use and compares this regularly with the time actually used. If the allotted time is about to be exhausted, the program terminates.

The job control (WFL) was set up as to recognize this case and start a new job to continue the interrupted simulation. The WFL is also able to recover from most system crashes. This allowed to run simulations for days or even weeks without any human interference.

II.1.3 Counting and Plotting Procedures

Two procedures were introduced to give the user of the program some feeling of whether or not the simulation was on the right way. The first of these ('counting procedure') simply counts the number of particles in each conventional unit cell and prints the counts. The second procedure ('plotting procedure') transforms all particles' positions into one unit cell and produces a printer plot of the projections of these positions onto the three principal lattice planes (100), (110) and (111). This allows an optical check on how close the particles remain to their equilibrium positions and if the system still resembles a solid.

II.2 Computing Forces and Potential

Since the potential function and its derivatives are given in form of a table, computing forces and potential requires to interpolate in this table. A second order Lagrange interpolation was chosen for this.

II.3 Program Verification

II.3.1 Harmonic Potential Tests

One test of the MD program was to simulate a two particle system with a harmonic pair potential. The influence of the PBC was eliminated in this case by making the box much larger than the separation of the particles and the range of the potential. This reduced the system to a true two particle problem, which can easily be solved analytically for a harmonic potential. The simulated trajectories could then be compared directly with the analytical solutions.

The harmonic potential allows a rigorous check of the interpolation. Since for a second order polynomial interpolation the interpolating function must be identical with the tabulated harmonical potential function, the results must be independent of the steplength of the table. This was verified.

II.3.2 Comparing with Results Reported for Rb

As a further test one of the computation of MSD in Rb, a = 5.739 Å, reported by Shukla and Mountain† was repeated. For a temperature of 279 K their result can be written as $\langle(\Delta r)^2\rangle/(a^2\ T) = 5.832\times10^{-5}$ /K. Our program produced for the same potential and almost the same temperature (271 K) the result $\langle(\Delta r)^2\rangle/(a^2\ T) = 5.952\times10^{-5}$ /K, which is about 2 % higher than Shukla and Mountain's. However we used a smaller time increment and integrated over a larger number of time steps. The resulting MSD tends to increase with decreasing time increment and increasing number of simulated time steps. This explains the small difference and our result is probably closer to the 'true' value.

† R. C. Shukla and R. D. Mountain, Phys. Rev. B, 25 (1982) 3649