MONTE CARLO CALCULATIONS OF THE RADIAL DISTRIBUTION FUNCTIONS FOR A DENSE PROTON-ELECTRON PLASMA

27.5.65

A.A. BARKER, B.Sc. (HONS). ADELAIDE

Mathematical Physics Department

University of Adelaide

Submitted August 1964

CONTENTS

SUMMARY

The radial distribution functions $g_{ab}(r)$ are computed for a neutral proton-electron equilibrium plasma of number density 10^{18}e/cc for temperatures of 1.35×10^{5}°K, 2×10^{4}°K and 10^{4}°K. A Monte Carlo technique, similar to that used by Wood and Parker in the study of fluids, is applied to a cell of 32 particles, the periodic boundary condition being used to reduce surface effects. The electrostatic energy due to the Coulomb interactions between particles is calculated analogously to the approach used by Madelung for calculating the cell energy in crystals. The computed radial distribution functions are compared with those obtained by the Debye-Hückel theory, the results of Broyles and Carley, and calculations done by Villars. Preliminary investigations at temperatures about 10^{5}°K indicated a slow convergence of the system to an equilibrium state, and this probably explains why the results disagreed strongly with the Debye-Hückel theory. The approach to equilibrium is interesting in that if this is studied in detail, useful information about irreversible processes can be obtained, but in this work we are more interested in obtaining the equilibrium $g_{ab}(r)$. Later results, even after sacrificing some accuracy to reduce computer time to comply with available

funds, confirmed that the convergence was still very slow at the lower temperature considered (10^4 °K), thus suggesting that a complete calculation would be extremely expensive. However, the approach shows considerable promise if enough computing time is available, as it can be applied to a wide range of densities and temperatures, and can thus be used for transition stages which are, in general,not covered by other methods, e.g. transition in a gas from the neutral to the ionized state. The run at 10^4°K with number density of 10^{18}, fell into this category, and confirmed the prediction of Saha's equation that for hydrogen isotopes, the gas is suddenly ionised at about 10^4°K. This run, being the longest, was the most important, and indicated that the equilibrium radial distribution functions at this temperature and density may become oscillatory. Such a conclusion, however, needs further confirmation, as the radial distribution functions are closely dependent on whether the system is in equilibrium, and in the above case it is unlikely that equilibrium has been attained. As mentioned to confirm the final form of $g_{ab}(r)$ will prove extremely expensive. However, should sufficient funds become available at some future date, consideration will be given to such a calculation.

I hereby declare that this thesis contains no material which has been accepted for the award of any other degree or diploma in any University, and that to the best of my knowledge and belief, the thesis contains no material previously published or written by any other person, except where due reference is made in the text.

A.A. BARKER

ACKNOWLEDGEMENTS

Throughout this work the author has been greatly stimulated and encouraged by Professor H.S. Green, with whom many helpful discussions have been held. The author is also particularly grateful to Dr. P.W. Seymour for numerous suggestions at opportune moments, and indeed wishes to acknowledge useful discussions with most members of the Adelaide University Mathematical Physics Department. Further this work could not have been completed without the generous grant from the Australian Institute of Nuclear Science and Engineering for computing. The computation was performed at the Weapon's Research Establishment, Salisbury, S.A., and the help of Barry McDowell and Alex Dunkeld with computational details was invaluable.

A.A. BARKER

LIBRARY OF THE UNIVERSITY OF ARKANSAS

I INTRODUCTION

1.1 The Radial Distribution Function

The importance of molecular distribution in fluids has by now been well established by several authors. Born and Green [1], Kirkwood [2], Yvon [3], Mayer [4] and Bogoliubov [5] have all obtained equations by which the fundamental variables of a fluid, such as the equation of state and thermodynamic variables, can be calculated in terms of the molecular radial distribution function of the system. The radial distribution function $g_{ij}(r)$ between a reference molecule of type i and other molecules of type j can be defined by $g_{ij}(r) = \rho_j(r)/\rho_{oj}$, where ρ_{oj} is the average number density of j type particles, and $\rho_j(r)$ is the actual number density of j type particles at a distance r from the i^{th} particle. Alternatively $\rho_{oj}\, g_{ij}(r)d^3x$ gives the probability of finding a particle of type j in the element of volume d^3x in the neighbourhood of a point $\underline{x}_j$ when a particle of type i is at $\underline{x}_i$, and $|\underline{x}_j - \underline{x}_i| = r$. To illustrate some of the properties of g(r) consider a liquid. In a simple liquid g(r) has an appearance given by Fig 1.1. It is zero for $r<a$, because at distances of closer approach there exists a large repulsive force between the molecules, and they are often considered as rigid spheres of diameter a. As $r\to\infty$ $g(r)\to 1$ since a particle will not affect the number density at large distances. The peaks define so called 'coordination

shells' where there is a marked preference for particles to accumulate, and this is due to the particles being excluded from r = o to a, and partly because such a configuration gives rise to the lowest potential energy of particle interactions. The form of g(r) is strongly temperature and density dependent, and for liquids the form has been experimentally obtained for particular temperatures and densities by scattering of X rays.

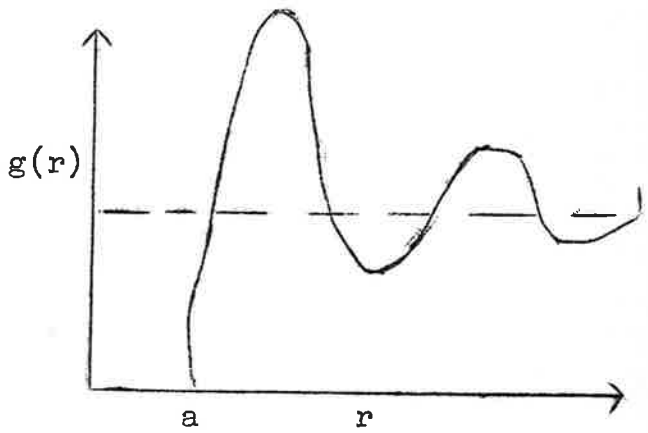


Fig 1.1
A typical radial distribution function for a simple liquid.

1.2 Early work on liquids

Most of the early work done by the authors of references [1] - [5], is concerned with applying the microscopic approach to liquids, and they generate integral equations, or a heirarchy of equations satisfied by the molecular distribution function. Attempts to solve these integral equations using various approximations were first made by Kirkwood [6] and A.E. Rodriguez and A.G. McLellan [7], who obtained closed integral equations for the radial distribution function using the superposition approximation, and then solved these numerically. The results showed that the theory gives the

correct qualitative descriptions of the radial distribution function, but that the quantitative results have significant errors, especially in some of the thermodynamic functions. In an effort to improve the agreement, Kirkwood, Lewinson and Alder [8] extended the application to molecules interacting with Lennard-Jones potential but still found a 10 to 20 per cent discrepancy with experiment. At about the same time, a somewhat different approach to the theory of liquids, originally proposed by Lennard-Jones and Devonshire [9], was being developed. This theory, called the cell theory, or Free Volume theory of liquids, has since been expanded by many authors [10], and Kirkwood in an early paper [11] has shown how the theory is related to his theory. However, in 1953, Metropolis et al [12] applied a new computing technique developed in the late 40's called the Monte Carlo method to solve this problem. This technique is suitable for calculation of properties of any substance which may be considered composed of individual interacting molecules, and although it is often used to solve integral equations, it gives a much better idea of the physical processes when it is applied to molecular problems, and is hence used extensively in transport problems where the Boltzmann integral equation is difficult to solve. The approach proved immediately successful, and Wood and Parker [13], in extending the method to molecules interacting with a Lennard-Jones potential, obtained results in excellent

agreement with the experimental results. This concept also suggested another approach introduced by Alder and Wainwright [14], which they call 'Molecular dynamics'. In this approach the computer actually follows the particles, and this proves particularly useful in checking critical points, such as phase transitions and some transport phenomena obtained in the previous calculations. Recently some attention has reverted to solving the integral equations by new techniques. Percus and Yevick [15] obtain slightly different equations using a collective coordinates approximation, and their equation has been solved by Broyles [16]. Broyles also solves an integral equation obtained via the Convolution Hypernetted Chain approximation [17] which uses the Mayer cluster expansion, and in [18] he compares the results of using Born and Green's equation with Perkus and Yevick's and the Convolution Hypernetted Chain approximation equation, and then compares them all with the accurate Monte Carlo calculations, finding that the Perkus Yevick equation gives best agreement.

1.3 Extension to Plasmas

H.S. Green [19] has extended his theory to plasmas, and obtained integral equations which he shows have the long established Debye Huckel radial distribution function as a first approximation. Green also finds that corrections from higher approximations are important, even at fairly low

densities. Recently Villars [20] has solved Green's equations and obtained radial distribution functions for a metallic plasma. However he experienced considerable difficulty with convergence at low temperatures. Also Broyles [21] has extended the Perkus Yevick and Convolution Hypernetted Chain approximations to long-range forces, and obtained radial distribution functions for a classical electron gas. At the present time however, the accuracy of the above solutions is in some doubt, and in this author's work it is intended to extend the accurate Monte Carlo method and obtain radial distribution functions for a proton-electron gas with specified number density and temperature.

Chapter 2 gives the procedure appropriate to this extension, and the theory behind it, and necessarily includes some of the work previously reported by Wood and Parker. Using this theory numerical calculations were then carried out for a dense plasma, and the computing details, with listing of the program, etc., are given in Chapter 4, which is really an oversize appendix. In Chapter 3 the results are compared with the results of Debye-Huckel, and with those of Villars as far as is possible. Differences are discussed and conclusions drawn.

REFERENCES TO CHAPTER I

[1] Born, M. and Green, H.S. Proc. Roy. Soc. A188 p.10 (1946)

[2] Kirkwood, J.G. Jour. Chem. Phys. 3 p300 (1935), 7, p919 (1939)

[3] Yvon, J. 'Fluctuations en Densité; La Propagation et la Diffusion de la Luminiére. (Herman and Cie) (1937)

[4] Mayer, J.E. Jour. Chem. Phys. 15 p187 (1947)

[5] Bogoliubov, N.N. 'Problems of Dynamical Theory of Statistical Physics' (English translation by E.K. Core Air Force Cambridge Research Centre, Publication TR-59-23)

[6] Kirkwood, J.G. and Boggs, E.M. Jour. Chem. Phys. 9 p514 (1941); 10 p394 (1942)

[7] Rodriguez, A.E., and McLellan, A.G. Proc. Roy. Soc. A196 p73 (1948); Proc. Roy. Soc. (London) Ser. A210 p509 (1951).

[8] Kirkwood, J.G., Levinson, V.A. and Alder, B.J., Jour. Chem. Phys. 20 p929 (1952)

[9] Lennard-Jones, J.E., and Devonshire, J.S., Proc. Roy. Soc. 163A p53 (1937).

[10] Chung, H.S., and Dahler, J.S., Jour. Chem. Phys. 37 p1620 (1962).

[11] Kirkwood, J.G., Jour. Chem. Phys. 18 p380 (1950)

[12] Metropolis, N.M., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E., Jour. Chem. Phys. 21 p1087 (1953).

[13] Wood, W.W., and Parker, F.R., Jour. Chem. Phys. 27 p720 (1957)

[14] Alder, B.J. and Wainright, A.W., Jour. Chem. Phys. 31 p459 (1959)

[15] Percus.,J.K., and Yevick., G.J., Phys. Rev. 110 p1 (1958)

[16] Broyles, A.A., Jour. Chem. Phys. 33 p 1068 (1961)

[17] Broyles, A. A., Jour, Chem. Phys. 34 p359 (1961)

[18] Broyles, A. A., Jour. Chem. Phys. 37 p2462 (1962)

[19] Green, H. S., Nuclear Fusion 1 p69 (1961)

[20] Villars, D.S., Physics of Fluids 6 p 745 (1963)

[21] Broyles, A.A., Sahlin, H.L., and Carley, D.D., Phys. Rev. letters 10 p319 (1963).

II THE METHOD

2.1 Procedure

2.1 a) General. The procedure adopted to find the radial distribution function for a plasma is considered before presenting the theory which justifies it, for this greatly enhances the understanding and presentation of the theory. However, because of this approach some points in this section appear incomplete, and in such cases some forward referring to the next section is resorted to, where the points will be more fully discussed and justified.

A system composed of N individual particles is considered confined in a volume V at a temperature T. The particles are assumed to obey classical statistics (see 2.2b). Further, in the interest of tractability, the particles are assumed to have spherically symmetric potential fields, and to interact with the Coulomb potential. Chapman and Cowling [1], indicate the enormous complexity when the condition of spherical symmetry is relaxed in the least degree. Subject to the above assumptions, the method is not restricted to any range of temperature or density, although as pointed out in the conclusion, the method is much more effective in specific regions.

In order to reduce the problem to a feasible size for numerical work, it is necessary to consider only a finite number of particles N. In this calculation N=32, being

comprised of 16 electrons and 16 protons. Such a choice is based on:- (a) convenience, for then a cube may be further split into eight compartments containing four particles each, and preliminary investigations made on these segments; (b) comparison, as many previous calculations have been done using N=32; (c) accuracy, as this value of N seems sufficient to accurately describe the system [2].

The 32 particles are placed in a unit cube of volume $V = L^3 = 1$ cubic cell unit, (it proves convenient to work in cell units or mesh units as defined in Appendix I). Since the computer can only deal with a finite number of digits k, the coordinates x_{i1}, x_{i2}, x_{i3} of the i^{th} particle, must be represented by a number $0 \leq x_{i\alpha} \leq 1$ where $x_{i\alpha}$ has a number of digits $\leq k$. Effectively this divides the unit cube into a fine mesh, and so will convert the number of distinct configurations (previously an integral) into a finite sum. The initial $C_{(j_1)}$ configuration's coordinates can be chosen arbitrarily by the use of random numbers, or alternatively the particles placed in a systematic fashion in positions which previous results have indicated will lead quickly to an equilibrium configuration. [The Random Numbers are generated by a technique given in 4.4]

Another configuration $C(k_r)$ is determined from $C(j_r)$ as follows. The coordinates of <u>one</u> of the particles are changed by a small random amount $\underline{\delta}_x$ such that $|\underline{\delta}_x| < \delta$,

i.e. a random $\underline{\delta}_{\underline{x}}$ is formed on the interval $-\delta_0$ to δ_0 and added to the old coordinates, so $x \to x + \delta_x$, $y \to y + \delta_y$, $z \to z + \delta_z$. (The choice of δ_0 is considered in section 2.2(c)). In general the molecule which suffers this displacement may either be chosen randomly or in a systematic fashion as in this calculation, where the particles are all labelled. The configuration which is the next in the series $C(j_{r+1})$ is either $C(j_r)$ or $C(k_r)$. To decide which the potential energy EP of $C(j_r)$ and ET of $C(k_r)$ are calculated (This energy calculation is considered in 2.1(c)) and a random number RAND is chosen on the interval $0 \leq RAND \leq 1$; if $0 \leq RAND \leq \frac{1}{1+e^{\beta(ET-EP)}}$ the configuration $C(j_r)$ is taken, if $\frac{1}{1+e^{\beta(ET-EP)}} < RAND \leq 1$ the configuration $C(k_r)$ is taken. Such a choice (as is shown in 2.2(a)), ensures that the relative frequency of a configuration $C(j)$ with potential energy $V(j)$ in the sequence of configurations is asymptotically $e^{-\beta V(j)}$. As these configurations are generated, and all 32 particles moved in turn, a store is kept of the pairing energy between any two particles, and also of the distance between any two particles within a limit of PDL (chosen to be a maximum distance without increasing computing time too much see 4.3). The pairing energies are stored so the energy of a configuration can be computed swiftly, while the storing of the pair distances

enables an estimate of the radial distribution function for like particles via

$$g_L(r) = \frac{\Delta N_L}{2\pi\, r^2\, \Delta r\, N^2\, n} \quad , \quad \text{and for unlike}$$

particles via

$$g_U(r) = \frac{\Delta N_U}{2\pi\, r^2 \Delta r\, N^2\, n} \quad ; \qquad (2.1)$$

Where N=32

n is the number of large iterations completed in the calculation, a 'large iteration' being defined when all 32 particles in the cell have been moved, or attempted to move.

ΔN_L is the number of like particles recorded at the distance r to r+Δr from the given particle, and ΔN_U the number of unlike particles in the same range.

2.1 (b) The Periodic Boundary Condition. So far only a basic cell of 32 particles has been considered. This means we are only considering a very small portion of the plasma, so to eliminate surface effects and obtain a representative sample the periodic boundary condition is employed. The basic cell is considered to be surrounded by exactly similar cells, in each of which the molecular configuration in the basic cell is identically repeated. Hence the appearance is similar to a crystal lattice, with repetition of the basic cell of 32 particles. The restriction of having identical cells, necessarily restricts some configurations

such as fluctuations in the number of particles in each cell. This restriction on the ergodicy of the problem has been shown to lead to only small errors for systems with $N \geq 30$ when dealing with fluids [2], but significant errors occur for low N or for very dense systems. However the identical repetition of cells is well suited to this computational technique and enables most of the surface effects to be eliminated, while still giving a practical solution.

2.1 (c) Calculation of the Potential Energy of a configuration.

The periodic boundary condition suggests the use of Crystal theory to evaluate the electrostatic potential energy between the particles. Born and Huang [3] show how to evaluate this electrostatic potential energy per cell, and call it the 'Madelung Energy' for a given crystal. For a plasma almost exactly the same technique as Born and Huang's can be used, but as this potential energy calculation is important some of their work will be repeated here.

Consider N particles in a cell. Let the position of the k^{th} particle in the cell be given by a position vector

$$\underline{X}(k) = \underline{a}_1 X_1(k) + \underline{a}_2 X_2(k) + \underline{a}_3 X_3(k), \text{ where}$$

X_1, X_2, X_3 are the components of $\underline{X}(k)$ along the three basic vectors of the cell $\underline{a}_1, \underline{a}_2, \underline{a}_3$, and in our case are orthogonal forming a face centred cubic lattice. Let the cells be specified by cell indices l_1 l_2 l_3 (these being in reference

to a zero cell chosen arbitrarily) and define a cell vector $\underline{X}(l) = l^1\underline{a}_1 + l^2\underline{a}_2 + l^3\underline{a}_3$. Hence the position vector of an arbitrary particle is represented by $\underline{X}(l) + \underline{X}(k)$, written $\underline{X}\begin{pmatrix} l \\ k \end{pmatrix}$. Also it is frequently useful to define a reciprocal lattice from the given lattice by defining basic cell vectors $\underline{b}^3 = \dfrac{\underline{a}_1 \times \underline{a}_2}{\underline{a}_3 \cdot \underline{a}_1 \times \underline{a}_2}$ etc., or $\underline{b}^\alpha \cdot \underline{a}_\beta = \delta_{\alpha\beta}$. From these reciprocal basic cell vectors $\underline{b}^\alpha$ (dimension of inverse length), a reciprocal lattice vector can be defined $\underline{y}(h) = h_1\underline{b}^1 + h_2\underline{b}^2 + h_3\underline{b}^3$. Thus $\underline{a}_1, \underline{a}_2, \underline{a}_3$ form a set of covariant base vectors, and $\underline{b}^1, \underline{b}^2, \underline{b}^3$ form a set of contravariant base vectors, and $\underline{X}(l).\underline{y}(h) = \sum_{i=1}^{i=3} l_i h_i$ = integer. These reciprocal lattice vectors are useful in specifying lattice waves satisfying the periodic boundary condition, since if $\underline{y}(h)$ is a reciprocal lattice vector, then $\exp[2\pi i \underline{y}(h).\underline{x}]$ is periodic in $\underline{x}$, and so a periodic function $f(\underline{x})$ can be expressed

$$f(\underline{X}) = \Sigma_h g(y)\, e^{2\pi i \underline{y}(h).\underline{X}(l)}$$

where the coefficients are given by $g(\underline{y}) = \int_0^1 dl^1 \int_0^1 dl^2 \int_0^1 dl^3 \left[f(l^1, l^2, l^3) e^{-2\pi i \underline{y}(h).\underline{X}(l)} \right]$ and $g(\underline{y})$ can then be varied to the form

$$g(\underline{y}) = \frac{1}{v} \int_{\text{cell}} f(\underline{x})\, e^{-2\pi i \underline{y}.\underline{X}}\, d\underline{X}. \qquad (2.2)$$

where v is the ~~cell~~ volume of the lattice cell, and $d\underline{X}$ denotes the volume element $dX_1 dX_2 dX_3$. The factor V^{-1} above arises from the Jacobian introduced by the change of integration variables from l^1, l^2, l^3 to X_1, X_2, X_3.

With the above notation, and assuming that the ions interact as point charges with a Coulomb potential (discussed later in 2.2(d)), the e.s energy for an arbitrary zero cell may be written as:-

$$E = \frac{1}{2} \sum_k \sum_{l'k'}{}' \frac{\xi_k \xi_{k'}}{\left| \underline{X}\begin{pmatrix} l' \\ k' \end{pmatrix} - \underline{X}\begin{pmatrix} 0 \\ k \end{pmatrix} \right|} \quad ; \qquad (2.3)$$

where ξ_k is the charge on a particle k

ξ_k' is the charge on a particle k'

k is an index running from 1 to N on particles in cell l=o,

k' is an index running from 1 to N on particles in cell l'

and l' is an index running from $-\infty$ to ∞.

The Σ' means that when l'=o, k' cannot equal k, as this would include self interaction energy in the sum. The factor $\frac{1}{2}$ appears in the formulae since the summation includes every interaction twice. Also notice that since a plasma is electrically neutral, then $\sum_k \xi_k = o.$ (2.4)

It is impractical to evaluate this energy by direct summation since the slow decrease of the coulomb interaction energy with distance means that theoretically the sum should be over an infinity of cells. To overcome this difficulty the energy is put in a quickly convergent series in the following manner. Rewrite (2.3) as

$$E = \tfrac{1}{2} \sum_k \xi_k \lim_{\underline{X}\to\underline{X}(k)} \left\{ \sum_{l'k'}' \frac{\xi_{k'}}{|\underline{X}\left(\begin{matrix}l'\\k'\end{matrix}\right)-\underline{X}|} - \frac{\xi_k}{|\underline{X}(k)-\underline{X}|} \right\},$$

then using Ewald's identity viz:-

$$\frac{1}{|\underline{X}\left(\begin{matrix}l'\\k'\end{matrix}\right)-\underline{X}|} = \frac{2}{\sqrt{\pi}} \int_o^\infty e^{-|\underline{X}\left(\begin{matrix}l'\\k'\end{matrix}\right)-\underline{X}|^2 \rho^2} \, d\rho ,$$

we have

$$E = \tfrac{1}{2}\Sigma_k \xi_k \lim_{\underline{X}\to\underline{X}(k)} \Sigma_{k'}\left\{ \frac{2}{\sqrt{\pi}} \xi_{k'} \int_0^\infty e^{-|\underline{X}\binom{1'}{k'} - \underline{X}|^2\rho^2} d\rho - \frac{\xi_k}{|\underline{X}(k)-\underline{X}|}\right\} \qquad (2.5)$$

The integral is then split into two quickly convergent parts by use of the Theta Transformation, viz:-

$$\frac{2}{\sqrt{\pi}} \Sigma_1 e^{-|\underline{X}(1)+\underline{X}(k')-\underline{X}|^2\rho^2} \equiv \frac{2\pi}{V} \sum_h \frac{1}{\rho^3} e^{-\pi^2/\rho^2 |\underline{y}(h)|^2 + 2\pi i \underline{y}(h).(\underline{X}-\underline{X}(k'))}$$

NOTE. This transformation is often referred to as the Theta Function identity, and can be derived quite generally in the following manner. Multiply

$$\Sigma_{\underline{k}} e^{2\pi i \underline{k}.\underline{x}} = \Sigma_{\underline{k}} \delta(\underline{x}-\underline{k}) \quad \text{by} \quad e^{-\pi\rho^2(\underline{x}+\underline{y})^2}$$

and integrate with respect to $\underline{x}$. This gives

$$\Sigma_{\underline{k}} \int e^{-\pi\rho^2(\underline{x}+\underline{y})^2 + 2\pi i \underline{k}.\underline{x}} d^3x = \Sigma_{\underline{k}} \int \delta(\underline{x}-\underline{k}) e^{-\pi\rho^2(\underline{x}+\underline{y})^2} d^3x$$

$$\text{hence} \quad \Sigma_{\underline{k}} \int e^{-\pi\rho^2(\underline{x}+\underline{y} - i\underline{k}/\rho^2)^2} e^{-\pi\underline{k}^2/\rho^2 - 2\pi i \underline{k}.\underline{y}} d^3x = \Sigma_{\underline{k}} e^{-\pi\rho^2(\underline{y}+\underline{k})^2}$$

$$\therefore \quad \frac{2\pi}{V} \sum_{\underline{k}} \frac{e^{-\pi\underline{k}^2/\rho^2 - 2\pi i \underline{k}.\underline{y}}}{\rho^3} = \sum_{\underline{k}} e^{-\pi\rho^2(\underline{y}+\underline{k})^2}.$$

Exchanging $\underline{k} = \underline{y}(h)$, $\underline{y} = (\underline{X}(k')-\underline{X})$, altering the π factors, and noticing that $\underline{y}(h) = \underline{X}(1)$ for a face centred cubic lattice, the form of the theta function which is usually applied to crystal physics is obtained. An alternative derivation is

given by Born and Huang [3], where the concept of reciprocal lattice vector expansions of a periodic function is stressed. The right hand side of this identity is a series that converges rapidly for small values of ρ, and the left hand a series that converges rapidly for large values of ρ. Thus dividing the range of integration of the integral in (2.5) and putting in the appropriate series we have

$$E=\tfrac{1}{2}\sum_k \xi_k \lim_{X\to \underline{X}(k)} \sum_{l'k'} \xi_{k'} \left\{ \frac{2\pi}{V} \int_0^R \frac{1}{\rho^3} e^{-\pi^2/\rho^2 |\underline{y}(g)|^2 + 2\pi i \underline{y}(h).(\underline{X}-\underline{X}(k'))} d\rho \right.$$

$$\left. + \int_R^\infty \frac{2}{\sqrt{\pi}} e^{-|\underline{X}\left(\begin{smallmatrix} l' \\ k' \end{smallmatrix}\right) - \underline{X}|^2 \rho^2} d\rho - \frac{\xi_k}{\underline{X}(k) - X} \right\}$$

The term $l'=o$ may be neglected in the first term (indicated by a prime) because of (2.4) and changing the order of integration and summation, with summation now over reciprocal lattice points in the appropriate case, this becomes:-

$$E=\tfrac{1}{2}\sum_k \xi_k \lim_{X\to \underline{X}(k)} \left\{ \int_R^\infty \frac{2}{\sqrt{\pi}} \sum_{l'k'} \xi_{k'} \exp\left[-|\underline{X}\left(\begin{smallmatrix} l' \\ k' \end{smallmatrix}\right) - \underline{X}|^2 \rho^2 \right] d\rho - \frac{\xi_k}{|\underline{X}(k)-\underline{X}|} \right.$$

$$\left. + \int_0^R \frac{2\pi}{V\rho^3} \sum_{k'h}{}' \exp\left(-\frac{\pi^2}{\rho^2} |\underline{y}(h)|^2 + 2\pi i \underline{y}(h).(\underline{X}-\underline{X}(k')) \right) d\rho \right\}.$$

The first term of the expression can be written in terms of a standard integral called the Error function, defined by $E(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$, since then $\frac{R}{Rt}(1-E(Rt)) = \frac{2}{\sqrt{\pi}} \int_R^\infty e^{-|t|^2\rho^2} d\rho$.

The integration of the second term can also be carried out

between the required limits to give:-

$$E=\tfrac{1}{2}\Sigma_k\xi_k \lim_{\underline{X}\to\underline{X}(k)} \left\{ R \sum_{1'k'} \xi_{k'} \frac{1}{R|\underline{X}\binom{1'}{k'}-\underline{X}|}\left[1-E(R|\underline{X}\binom{1'}{k'}-\underline{X}|)\right] - \frac{\xi_k}{|\underline{X}(k)-\underline{X}|} \right.$$

$$\left. + \frac{\pi}{V}\sum_{k'}\xi_{k'}\sum_{h}{}' \frac{\exp\left(-\frac{\pi^2}{R^2}|\underline{y}(h)|^2\right)\exp[2\pi i\underline{y}(h).(\underline{X}-X(k'))]}{\pi^2|\underline{y}(h)|^2} \right\}$$

In taking the limit, it can be seen that apart from the second term and the term $\binom{1'}{k'}=\binom{0}{k}$, all terms are regular functions of $\underline{X}$ in the neighbourhood of $\underline{X}(k)$, and hence in these terms one can directly replace $\underline{X}$ by $\underline{X}(k)$. The prime on the summation means the term $\binom{1'}{k'}=\binom{0}{k}$ is excluded. Thus

$$E = \frac{R}{2}\sum_{1k'k}{}'\xi_k\xi_{k'}\frac{1}{R|\underline{X}\binom{1'}{k'}-\underline{X}(k)|}\left[1-E(R|\underline{X}\binom{1'}{k'}-\underline{X}(k)|)\right]$$

$$+ \frac{\pi}{2V}\sum_{kk'}\xi_k\xi_{k'}\sum_{h}{}'\exp\left(\frac{\pi^2|\underline{y}(h)|^2}{R^2}\right)\exp[2\pi i\underline{y}(h).(\underline{X}(k)-\underline{X}(k'))]$$

$$+ \frac{1}{2}\Sigma_k\xi_k^2 \lim_{\underline{X}\to\underline{X}(k)}\left\{ \frac{R}{|\underline{X}(k')-\underline{X}|R}\left[1-E(R|\underline{X}(k')-\underline{X}|)\right] - \frac{1}{|\underline{X}(k)-\underline{X}|}\right\}$$

This last term then has the form

$$\lim_{\zeta\to 0}\left\{\frac{R}{R\zeta}[1-E(R\zeta)]-\frac{1}{\zeta}\right\} = \lim_{\zeta\to 0}\frac{2}{\sqrt{\pi}}\frac{1}{\zeta}\left\{\int_{R\zeta}^{\infty}e^{-\rho^2}d\rho - \frac{\sqrt{\pi}}{2}\right\}$$

$$= \lim_{\zeta \to 0} \frac{2}{\sqrt{\pi}} \frac{1}{\zeta} \left\{ - \int_0^{R\zeta} e^{-x^2} dx \right\}$$

$$= \lim_{\zeta \to 0} \frac{2}{\sqrt{\pi}} \frac{1}{\zeta} \left\{ - \int_0^{R\zeta} [1 - x^2 + \frac{x^4}{2} - \ldots] dx \right\}$$

$$= - \frac{2R}{\sqrt{\pi}}$$

Now putting $D = R|\underline{X}(k') - \underline{X}(k)|$ we obtain

$$E = \frac{R}{2} \sum'_{k'k} \xi_k \xi_{k'} [1 - E(D)]/D - \frac{R}{\sqrt{\pi}} \sum_k \xi_k^2 \qquad (2.6)$$

$$+ \frac{\pi}{2V} \sum_{kk'} \xi_k \xi_{k'} \sum'_h \frac{\exp(-\pi^2 |\underline{y}(h)|^2 / R^2) \exp[2\pi i \underline{y}(h) \cdot (\underline{X}(k) - \underline{X}(k'))]}{\pi^2 |\underline{y}(h)|^2}$$

NOTE. This equation divides the Energy into calculations involving a) short range interactions, which are accounted for in term 1, and b) long range interactions, which are accounted for in term 3 (which since the lattice is periodic, sums over the reciprocal lattice vectors of the unit cell). However, it is important to notice that the energy associated with term 1 is NOT the energy of the short range interactions, as term 2 also involves some short range energy, the amount depending on the parameter R. Thus the above formulae does not enable the calculation of the energy in two distinct parts, but gives a practical computation of the total electrostatic energy of a cell. The energy will be in ergs if ξ_k is in e.s.u., and R is in $(cm)^{-1}$. This ensures that the last term is dimensionally correct, as $\underline{y}(h)$, the reciprocal lattice vector has dimension of inverse length, and V is the volume of the cell.

In crystal theory R is usually chosen of order 1/(nearest ion-ion distance) for quick convergence of both series. Based on this, for the plasma R was initially set approximately equal to 1/(nearest ion-ion distance), however experience has shown that a better indication of the order of R is given by 1/(Debye Shielding distance). The exact value chosen for R is subject to other considerations also, such as the accuracy to be obtained for given computing time, and will be discussed in detail in section 2.3. As only energy differences count, and R remains constant for a particular run, the energy actually calculated in cell units (see Appendix for conversion factors) is given by

$$E = \frac{R}{2} \sum'_{kk'l} C_k C_{k'} \frac{1}{D} [1-E(D)] + \frac{1}{2\pi} \sum_{kk'} C_k C_{k'} \sum'_{l} \frac{\exp(-\pi^2 |\underline{X}(l)|^2/R^2)}{|\underline{X}(l)|^2}$$

$$\times \cos 2\pi l_1(X(k)-X(k')) \times \cos 2\pi l_2(Y(k)-Y(k')) \times \cos 2\pi l_3(Z(k)-Z(k')) \quad (2.7)$$

Where C_k and $C_{k'}$ are +1 for a proton, and -1 for an electron, and the exponential term has been expanded in terms of real and imaginary parts, and the imaginary parts neglected, as only the real contributions to the energy are required.

The calculation of the e.s. cell energy via (2.7) is still very time consuming, and so a careful analysis is made in section 2.3 as to the choice of R, the termination of the two series, the accuracy, and other factors which affect the computing

time. The rest of the program is built around this energy calculation, and carries out the general procedure explained earlier. Also a significant part of the program is necessarily concerned with the input of data and the output of results. A full listing of the program, and explanation of what each part does is given in Chapter 4.

2.2 Theory

2.2 a) General. Before passing on to considerations of accuracy and choice of variables for the computation, it is perhaps necessary to give a brief outline of the theory underlying the procedure mentioned in the previous section. The idea of applying the Monte Carlo method to compute radial distribution functions was proposed by N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H.Teller, and E. Teller [4] and followed by Wood and Parker [5]. The following theory is based on the above two papers. A petite canonical ensemble is constructed from a canonical ensemble (i.e. reducing the continuum of configurations to a countable set of configuration,) by considering only a finite number of digits in specifying the coordinates of a particle. This effectively divides the unit cell into a very fine mesh, such that the positions of the molecules are specified by a single number associated with each mesh cell. This division is inherent in numerical calculation, and it is physically reasonable that a sufficiently fine

subdivision will give results indistinguishable from the continuum of configurations. To calculate the equilibrium value of any quantity of interest F, then

$$\bar{F} = \frac{\Sigma_j F_j \exp(-E_j/kT)\, d^{3N}p\, d^{3N}q}{\Sigma_j \exp(-E_j/kT)\, d^{3N}p\, d^{3N}q}, \quad \text{where } j$$

runs over all configurations. Also since in the interest of tractability velocity independent forces have been assumed, the momentum integral may be separated analogously to the following

$$\int e^{-\beta(KE+V)}\, d\underline{p}\, d\underline{x} = \int e^{-\beta KE} d\underline{p} \int e^{-\beta V} d\underline{x}$$

$$= \int e^{-\frac{\beta}{2m} p^2} d\underline{p} \int e^{-\beta V} dx$$

$$= \left(\frac{2\pi m}{\beta}\right)^{3N/2} \int e^{-\beta V} d\underline{x}, \text{ and hence } \bar{F}$$

can be written

$$\bar{F} = \frac{\Sigma_j F_j \exp(-V(j)/kT) d^{3N}q}{\Sigma_j \exp(-V(j)/kT\, d^{3N}q} \qquad (2.8)$$

Now even in the case where N is reasonably small, it is impracticable to carry out the multidimensional sum (or integral) by usual numerical methods, and so the Monte Carlo method is used. The Monte Carlo method of evaluating many-dimensional integrals consists of integrating over a

random sampling of points instead of over a regular array of points, (An authoritive text on the application of the Monte Carlo method is that by Herman Kahn [6].) In this case, if the Monte Carlo technique was applied in a straightforward manner, a random configuration would be chosen, and then weighted by $\exp(-\beta V(j))$. However, a more practicable method is to choose configurations with a probability $\exp(-\beta V(j))$, and wieght them evenly. Thus as Wood and Parker put it, 'the object is to generate a Markov chain in which asymptotically each state k recurs with a frequency proportional to the Boltzmann factor $\exp(-\beta V(j))$ for that state'. For then the average over the chain of any function of the configuration state, such as F_k, will converge to the corresponding petite canonical ensemble average of the same quantity, as the chain length increases.

The manner in which such a sequence is generated has been discussed in the previous section 2.1(a). To show that the relative frequency of C(j) in the sequence of configurations thus constructed is asymptotically $\exp(-\beta V(j))$, suppose the probability that j_r should have the value j is $p_r(j)$. Let C(k) be a configuration obtained from C(j) in the manner described in the procedure. Then C(k) belongs to a set of configurations accessible from C(j) by changing the coordinates of one particle by amounts less than δ_o. If E(j) denotes this ensemble (or set of configurations), then $k \epsilon$ E(j).

Let Q be the number of distinct configurations in each of the ensembles E(j) and E(k). Now the probability of moving from state j_{r+1} back to state j is $p_{r+1}(j)$, and we have

$$Q\, p_{r+1}(j) = p_r(j) \sum_{k\epsilon E(j)} [1-\alpha(j,k)] + \sum_{j\epsilon E(k)} p_r(k)\alpha(k,j) \qquad (2.9)$$

where

$$\alpha(j,k) = \{1 + \exp[\beta V(k) - \beta V(j)]\}^{-1} , \qquad (2.10)$$

= probability that C(j) will be changed to C(k) by displacing a molecule in C(j).

$\sum_{k\epsilon E(j)}$ is the summation over all values of k such that $k\epsilon E(j)$.
Now the relative frequency of configuration C(j) is asymptotically proportional to p(j). But

$$p(j) = \lim_{s\to\infty} s^{-1} \sum_{r=1}^{s} p_r(j) .$$

Hence from (2.9)

$$Qp(j) = p(j)\, [Q - \sum_{k\epsilon E(j)} \alpha(j,k)] + \sum_{j\epsilon E(k)} p(k)\alpha(k,j).$$

i.e. $$p(j) \sum_{k\epsilon E(j)} \alpha(j,k) = \sum_{j\epsilon E(k)} p(k)\alpha(k,j). \qquad (2.11)$$

This means that the probabilities must satisfy the condition of microscopic reversibility. Also in this set of simultaneous equations to determine the p(j), if we replace $\alpha(j,k)$ with the expression (2.10), the solution is

$p(j) = A\exp[-\beta V(j)]$, where A is independent of j,

and this is the property of the sequence $C(j_1)$, $C(j_2)$,... we wished to prove. Notice that the above proof assumes that the sequence is ergodic, and this will only be so if all particles may reach any point in configuration space. This is the case here as there will always be a non-vanishing probability of moving from a given configuration to any other in a finite number of moves, and this is because the potential energy is finite for all configurations. The finite nature of the potential energy follows because a cut-off is imposed on the coulomb potential well, so that particles approaching closer than the potential well cut-off distance A0 have constant potential energy. The choice of A0 is important, and is discussed fully in section (2.2d). Thus we have obtained a sequence which gives the petite canonical ensemble average of (2.8) in the form

$$\langle F \rangle = \lim_{S\to\infty} S^{-1} \sum_{r=1}^{S} F(jr) \qquad (2.12)$$

2.2 b) The distribution choice. There are other ways of assigning the probabilities $\alpha(j,k)$ so that equation (2.11) still has a solution $p(j)\alpha \exp[-\beta V(j)]$. However, H.S. Green has shown that the choice of $\alpha(j,k)$ given in (2.10) secures the most rapid convergence of the series (2.12). This choice is of considerable practical importance, since in the calculation it is necessary to replace (2.12) by

$$\langle F \rangle = S^{-1} \sum_{r=V+1}^{S} F(jr),$$ with suitably chosen values of V and S, and if a bad choice is made for $\alpha(j,k)$, then these values of R and S will be unnecessarily large. Effectively $\alpha(j,k)$ is a normalised Boltzmann distribution for particles allowed to move between two potential energy states V(j) and V(k), as from the Boltzmann distribution law the probability of a particle frequenting a particular state V(j) rather than V(k) is

$$p(j) = \frac{e^{-\beta V(j)}}{e^{-\beta V(j)} + e^{-\beta V(k)}}$$

$$= \frac{1}{1 + e^{-\beta(V(k)-V(j))}}$$

$$= \frac{1}{1 + e^{\beta(V(j)-V(k))}} \quad .$$

Fig 1 shows how this probability varies with temperature for some typical changes in potential energy between two states. If the state j has energy greater than the proposed new state k, then $\delta E = V(j) - V(k)$ will always be positive, and the probability of staying in j will always be less than 0.5. An energy change $\delta E = 0.1$ cell units

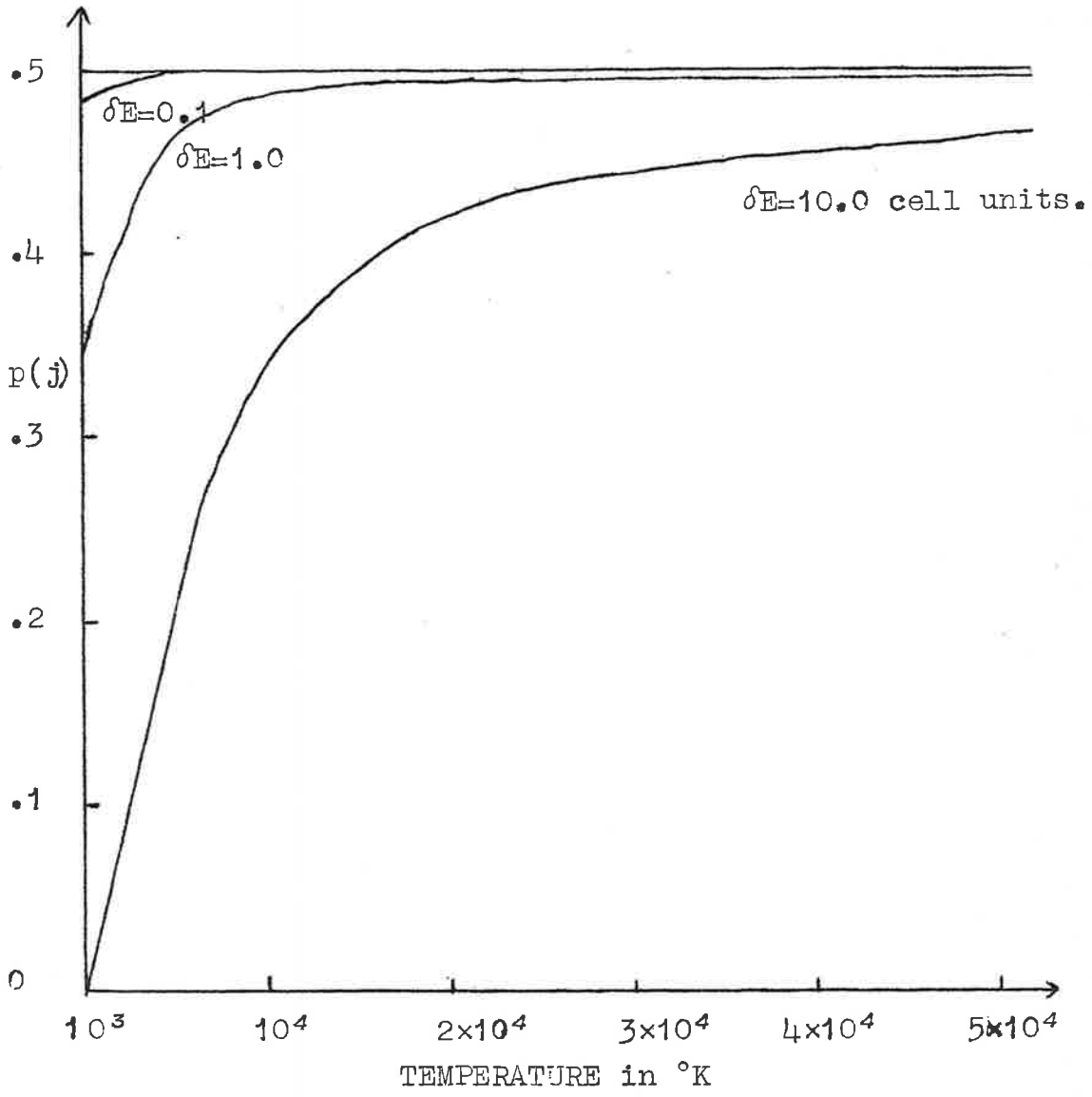


FIG 2.1 Showing the probability p(j) of a particle changing its energy by a positive amount δE at a temperature T.

is approximately equivalent to shifting a particle from 286 mesh units to 271 mesh units from another unlike particle. Similarly an energy change 1.0 corresponds to moving the unlike particle from 80 mesh units away to a lower energy position only some 70 units away, and an energy change of 10.0 to a movement from 12 to 9.5 mesh units. Hence for small energy changes it can be seen that one must go to temperatures below 10^{40}K before the probability of a transition differs significantly from 0.5, and above this temperature small energy changes will occur nearly randomly using the Monte Carlo method. Larger energy changes however, have a marked effect on the probability, even at temperatures as high as 10^{50}K. Results have shown that the size of δE depends on two main factors; (a) the size and direction of the particle movement $\underline{\delta}x$, which will be discussed in the next section, and (b) the position of the closest particle. It has been shown that a small change $\underline{\delta}x$ will incur a large change δE if another particle is only a distance of 12 mesh units away, but that it requires a much larger $\underline{\delta}x$ to cause the same δE when the other particle is 80 mesh units away.

Consider the 32 particles moving randomly about the unit cell, then the unnormalised probability that any one of the 16 positive particles will be within a distance of 100 mesh units of any one of the negative particles is given by

$$P = 16\times16\times\tfrac{4}{3}\pi \times100\times100\times100 = 10.7\times10^{8},$$ which

is much larger than the corresponding probability of particles being within 50 mesh units, when

$$P = 16\times16\times\tfrac{4}{3}\pi \times50\times50\times50 = 1.34\times10^{8}.$$

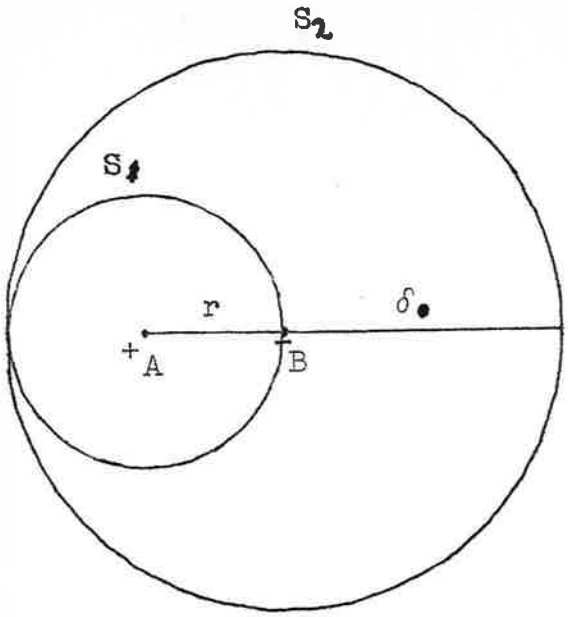
In calculations done on the computer at about 10^{5}°K, the particles distribute themselves roughly throughout the unit cell, and have an average interparticle distance of approximately 400 mesh units. There are nearly always two unlike particles separated by less than 100 mesh units, and in future two such particles will be referred to as '<u>linked</u>'. The calculations also show that most of the energy changes when a particle is moved are of order 0.1 cell units of energy, and hence statistics will affect these only rarely at high temperatures. It becomes almost a matter of chance at these high temperatures that two unlike particles will become closely linked, and as shown by P the closer the link, the less probable it is to occur from a completely random jump. In fact due to the manner in which the solution to the problem has been presented, close links occur even less than expected from P at high temperatures, for in 1000 iterations a particle only moves on approximately half the attempted moves, i.e. some 500 times, and in a random direction thus moving through about one eighth of the unit cell. During this random travel the particle has on the average a chance to link with only two oppositely charged particles, hence P should

equal $2\times16\times\frac{4}{3}\pi\times100\times100\times100$, equals 1.34×10^{8}. These close links have an important effect on the radial distribution function at low distances, but because they occur so rarely very long runs must be done before much weight can be put on g(r) for r<100 mesh units.

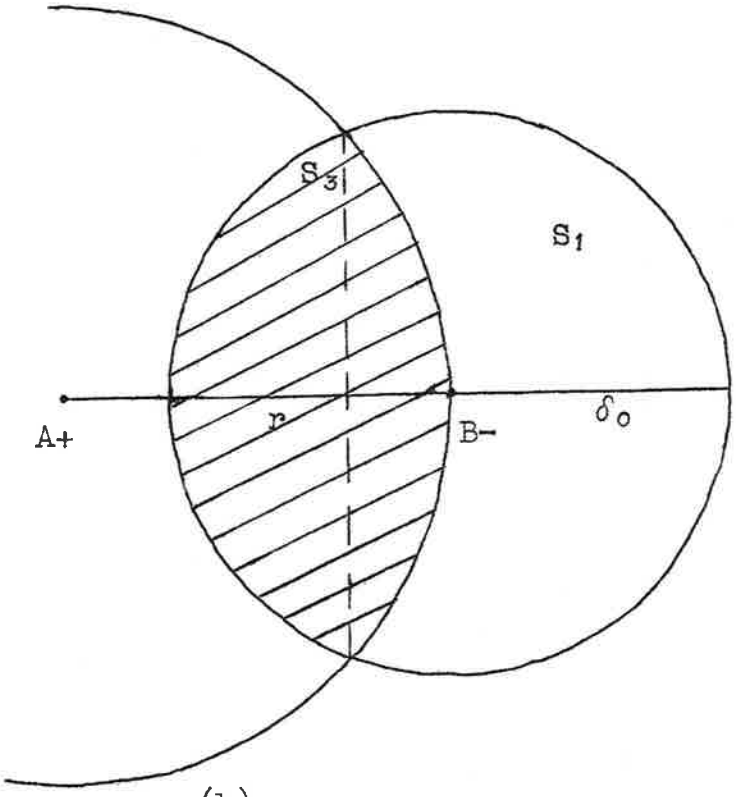
As the temperature is lowered however, the probability p(j) of moving to a close link, starts to become less random, and the effect of the statistics becomes apparent, close links occur much more frequently, and the radial distribution function assumes a reasonably smooth curve in the region 0 to 100 mesh units; though now long runs are needed before the system settles into equilibrium. In summary the temperature dependence of p(j) varies from allowing almost random motion at 10^{50}K to having a marked effect on movements at 10^{40}K. Another interpretation of how the Boltzmann statistics are built into this problem, which is quite useful in discussion of the results, is that each particle has associated with it an effective statistical radius, beyond which the effect of the statistical distribution is negligible, but within which the statistics appreciably influence particle movements. This effective statistical radius becomes larger as the temperature is lowered. The importance of the statistics (or the statistical distribution choice), will again be stressed when the results are discussed.

2.2 (c) Choice of δ_0. This determines the maximum change of molecular coordinates in the transition from one configuration to the next, and as already indicated, has an important influence on the rapidity of convergence of the method. H.S. Green has shown that a $\delta_0 \simeq \frac{L}{3N}$ has the right order of magnitude for most calculations. If δ_0 is too large, the energy/cell will fluctuate considerably with each new iteration, while on the other hand if δ_0 is too small, not enough samples of configuration space will be obtained in a reasonable time. There is perhaps an argument for letting δ_0 be large to start with, and as the system nears equilibrium, to decrease δ_0 to a smaller value.

The advantages and disadvantages of choosing a large δ_0 show up in the following considerations. Let two unlike particles A and B be a small distance r apart, and also for definiteness let this be at a low temperature where the statistics highly favour the formation of a pair, i.e. a transition of the particle from r to the ground state. Then if δ_0 is large, it will take a considerable time before a delta occurs which will allow the particles to move closer together, for if $\delta_0 \geqq 2r$ fig 2.2 (a) is obtained. Suppose the program allows B to undergo a random displacement $\underline{\delta}_x$ where $|\underline{\delta}_x| \leq \delta_0$, then B can move to any point in the sphere S_1, and the relative probability P(c) that it moves closer to A rather than further away from A is given by the ratio of the volume of S_1 to the volume $S_2 - S_1$.



(a)



(b)

FIG 2.2 Diagrams used in the calculation of an optimum maximum step length δ_0.

Thus $P(c) = \frac{4/3\ \pi\ r^3}{4/3\pi(\delta_o{}^3 - r^3)} = \frac{r^3}{\delta_o{}^3 - r^3}$, and

if $\delta_o = 2r$ as drawn in fig 2.2(a), then $P(c) = \frac{1}{7}$, so on the average, out of eight trial movements of B, on only one occasion will it move to a lower energy position closer to A. Since we are considering statistics which highly favour movements to lower energy the seven moves will not be allowed, and the move closer will be accepted. Even when A and B are a reasonable distance apart so $\delta_o < 2r$, the probability of B moving closer to A rather than further away will be given from fig 2.2(b) by the ratio of the shaded volume S_3 to the volume $S_1 - S_3$. Calculating S_3 as the volume of two caps then

$$P(C) = \frac{8\frac{r}{\delta_o} - 3}{8\frac{r}{\delta_o} + 3}$$

TABLE 1.

	Case (b) where $r \leq \frac{\delta_o}{2}$		Case (b) where $r > \frac{\delta_o}{2}$		
r =	$\delta_o/4$	$\delta_o/2$	δ_o	$2\delta_o$	$>>\delta_o$
$P(c)$ =	1/63	1/7	1/2.2	1/1.46	1/1

This shows that for $r = \delta_o/4$ that only one move out of an attempted 64 will, on the average, result in the particles coming closer together. Further because of the potential

well cut-off discussed in the next section, movements of particles closer than 4.2 mesh units will have the same energy as if they only moved to 4.2 mesh units apart. Hence δ_0 for low temperature work was chosen to have a value of 15 mesh units, as this means not an excessive number of moves has to be made before the particles have a chance to pair, and yet the particles can move through a reasonable volume in 100 iterations. However at high temperatures, where statistics only weakly influence movement to lower energies even when the particles are close together and movement is mainly random, a larger δ_0 would seem preferable.

2.2 (d) The potential well cut-off AO The choice of this variable, which limits the potential energy between interacting particles when they approach each other closer than a distance AO mesh units, turns out to be important if the degree of ionisation is to be considered. The actual value of AO chosen is based on the following considerations. The Bohr orbit theory shows that when an electron is in its ground state in a hydrogen atom, it has a potential energy of $-e^2/a$, and a kinetic energy of $e^2/2a$, where a is the first Bohr radius. The ionisation energy is the energy needed to remove the electron from its present energy of $-e^2/2a$ to the continuum or zero energy. Then it can be seen that the lowest potential energy possible, without any kinetic energy, will be the ionisation energy of $-e^2/2a$.

This potential energy corresponds to that of an electron at a distance 2a from the proton, and since this is the ionisation energy no lower potential energy is possible without kinetic energy being present also, and as this problem is concerned with potential differences, an electron and proton are considered as having their lowest interaction energy of $-e^2/2a$, whenever an electron comes closer than A0 = 2a to the proton.

The importance of this cut-off A0 becomes apparent IF it is said that 'pairing' occurs when two particles are closer than A0. This critereon for pairing means that the particles are in their lowest possible potential energy state with no kinetic energy, i.e. ground state, and so particles are not considered as paired when they are in excited states, such states have previously been referred to by saying the particles are 'linked'. In Saha's work, which is mentioned in some detail when the results are discussed, he considers particles paired only if they are in the ground state as above, but he does not allow for any excited states (except the continuum) to exist. This differs from the above case, where a continuum of excited states (or links between particles) are allowed to exist outside the ground state radius A0, as particles may move out from A0 in a completely random fashion, being modified only by the statistics. No quantum effects, or only allowing special energy levels when approaching particles near the ground state, are built into this method. This difference in the definition

of pairs - or neutral particles, and merely links between two charged particles must be kept in mind when comparison is made with Saha's work.

2.3 Accuracy

2.3 (a) The first term From equation (2.7) the first term E_1 is given by

$$E_1 = \frac{R}{2} \sum_{k=1}^{NA2} C_k \sum_{k'=1}^{NA2,} C_{k'} \sum_{NL=-\infty}^{\infty} \left(\frac{1.0-E(D)}{D} \right) , \text{ where NA2}$$

is the number of particles in the reference cell, and the similar periodic cells are labelled by NL. To terminate the inner series the energy contribution of particles interacting beyond a distance X_c is neglected; and this effectively considers interactions over only a finite number of cells NL. To estimate the error in cutting off at $X_c = \frac{D_o}{R}$, assume a rectangular distribution of particles with distance, for then the percentage error in E_1 is given by

$$\frac{\int_{D_o}^{\infty} \frac{1-E(D)}{D} \, 4\pi D^2 \, dD}{\int_{o}^{\infty} \frac{1-E(D)}{D} \, 4\pi D^2 \, dD} .$$

But

$$\int_{D_o}^{\infty} \frac{1-E(D)}{D}\ 4\pi D^2\ dD = \int_{D_o}^{\infty} \left(\frac{2}{\sqrt{\pi}}\ 4\pi D \int_{D}^{\infty} e^{-y^2}\, dy \right) dD$$

$$= 8\sqrt{\pi} \left\{ \left[\frac{D^2}{2} \int_{o}^{\infty} e^{-y^2}\, dy \right]_{D_o}^{\infty} - \int_{D_o}^{\infty} \frac{D^2}{2}\ e^{-D^2}\, dD \right\}$$

$$= 8\sqrt{\pi} \left\{ \left[\frac{D^2}{2} \left(\frac{\sqrt{\pi}}{2} - \int_{o}^{D} e^{-y^2}\, dy \right) \right]_{D_o}^{\infty} - \left[\frac{1}{4}\ D\ e^{-D^2} \right]_{D_o}^{\infty} + \int_{D_o}^{\infty} \frac{1}{4}\ D\ \frac{1}{-2D}\ e^{-D^2}\, dD \right\}$$

$$= -8\sqrt{\pi}\ \frac{D_o^2 \sqrt{\pi}}{4} \left[1-E(D_o) \right] + \frac{8\sqrt{\pi}}{4}\ D_o e^{-D_o^2} - \frac{8\sqrt{\pi}}{8} \left[\frac{\sqrt{\pi}}{2} - \frac{\sqrt{\pi}}{2}\ E(D_o) \right]$$

$$= -2\pi\ D_o^2 (1-E(D_o)) + 2\sqrt{\pi}\ D_o e^{-D_o^2} - \frac{\pi}{2}\ (1-E(D_o))$$

$$= 2\sqrt{\pi}\ D_o\ e^{-D_o^2} - (2\pi D_o^2 + \frac{\pi}{2})\ (1-E(D_o)).$$

Also $\int_{o}^{\infty} \frac{1-E(D)}{D}\ 4\pi D^2\, dD = -\frac{\pi}{2}$, so the percentage error in term one is $\left[2\sqrt{\pi}\ D_o\ e^{-D_o^2} - (2\pi\ D_o^2 + \frac{\pi}{2})\ (1-E(D_o)) \right] \times \frac{100}{-\frac{\pi}{2}}$.

TABLE 2.

D_o	1.0	2.0	2.5	2.7	2.75	3.0
Error %	4.4	.26	.03	.015	.01	.006

Hence a value of D_o of about 2.5 would be sufficient to determine E1 to three figure accuracy, this value is dimensionless as $D_o = RX_c$.

2.3 b) Speed Considerations The most time consuming of all the operations completed by the computer is the calculation of $E(D) = \frac{2}{\sqrt{\pi}} \int_o^D e^{-t^2} dt$. The number of these to be calculated will depend upon X_c, because all contributions with $|\underline{X}\left(\begin{matrix}l'\\k'\end{matrix}\right) - \underline{X}(k)| > X_c$ are neglected. Thus for optimum speed X_c is to be as small as possible.

2.3 c) Debye Huckel factors The calculation of the energy is essentially divided into two parts, in one individual interactions are considered, in the other a collective approach is used (second energy term). From Debye Huckel theory individual interactions predominate within the Debye Shielding distance λ_D. This suggested that to obtain reasonable accuracy, X_c should be made $\geq \lambda_D$. Table 3, compiled for a neutral plasma of density 10^{18} electrons/cc, gives some idea of λ_D in cell units as temperature varies.

TABLE 3

Temp °K	10^4	2×10^4	5×10^4	10^5	1.35×10^5	5×10^5
λ_D(cell units)	.194	.275	.435	.615	.715	1.375

2.3 d) Accuracy of the second term From equation (2.7) the second term E_2 is given by

$$E_2 = \frac{1}{2\pi} \sum_{k=1}^{NA2} C_k \sum_{k'=1}^{NA2} C_{k'} \sum_{NL=-\infty}^{\infty}{}' \left\{ \frac{\exp(-\pi^2 |\underline{X}(1)|^2/R^2}{|\underline{X}(1)|^2} \cos 2\pi l_1 (X(k)-X(k')) \right.$$

$$\left. \cos 2\pi l_2 (Y(k)-Y(k'))\cos 2\pi l_3 (Z(k)-Z(k')) \right\}$$

Let $X = 2\pi(X(k)-X(k'))$ $Y = 2\pi(Y(k)-Y(k'))$ $Z = 2\pi(Z(k)-Z(k'))$

Then performing the sum over NL, with NL(0,0,0) excluded, the following is obtained.

$$E_2 = \frac{1}{2\pi} \sum_{k=1}^{NA2} C_k \sum_{k'=1}^{NA2} C_{k'} \left\{ 2e^{-\pi^2/R^2}(\cos X+\cos Y+\cos Z) \right.$$

$$+ 2e^{-2\pi^2/R^2}(\cos X\cos Y+\cos X\cos Z+\cos Y\cos Z)$$

$$+ {}^{8}/_{3}\, e^{-3\pi^2/R^2}(\cos X\cos Y\cos Z)$$

$$+\tfrac{1}{2}\, e^{-4\pi^2/R^2}(\cos 2X+\cos 2Y+\cos 2Z)$$

$$+ \tfrac{4}{5}\, e^{-5\pi^2/R^2}(\cos X\cos 2Y+\cos 2Y\cos Z$$

$$+\cos Y\cos 2Z+\cos 2X\cos Z+\cos 2X\cos Y$$

$$+\cos X\cos 2Z)$$

$$\left. + \cdots\cdots\cdots \right\}$$

The magnitude of this term increases with R, and for large R many terms will need to be taken as convergence will be slow. However if R is less than π the series will converge rapidly

due to the exponential factor, and terms after the third one become negligible. To obtain a very rough idea of the error involved in neglecting higher terms, consider the percentage error given by terms 4,5 and 6 compared to terms 1,2 and 3. To evaluate this ratio in terms of R, the cos terms are replaced by $\frac{1}{\sqrt{2}}$. This gives as an estimate for the percentage error in E_2

$$100 \times \frac{\left\{ \frac{1}{2} e^{-4\pi^2/R^2} \left(\frac{3}{\sqrt{2}}\right) + \frac{4}{5} e^{-5\pi^2/R^2} 3 + \frac{4}{3} e^{-6\pi^2/R^2} \frac{3}{2\sqrt{2}} \right\}}{\left\{ 2e^{-\pi^2/R^2} \left(\frac{3}{\sqrt{2}}\right) + 2e^{-2\pi^2/R^2} \frac{3}{2} + \frac{8}{3} e^{-3\pi^2/R^2} \frac{1}{2\sqrt{2}} \right\}}$$

$$\simeq 25\, e^{-3\pi^2/R^2}$$

TABLE 4.

R (inverse cell units)	1	2	2.5	2.7	3.0	3.5
% Error in E_2	10^{-10}	.05	.3	.5	1	3

Although this calculation is very sketchy, the thorough calculation of the root mean square error is very complex, and it was found in practice that the above table was more accurate than might be expected, and quite sufficient, with the other tables, to obtain a working value for R.

2.3 (e) Summary To obtain optimum accuracy, or approximately so, the error in calculating the first term of the energy should equal the error in calculating the second term,

i.e. $E_1 \times$%error in $E_1 \simeq E_2 \times$%error in E_2 (2.13)

However, since it is the object of the program to calculate the equilibrium energy for a given temperature and density, it is difficult to know the values of E_1 and E_2. To obtain some indication of their sizes some preliminary calculations were performed using N=4. This was later found to be only a very approximate guide, as a box of 32 particles in equilibrium cannot be accurately constructed from 8 lots of 4 particles in equilibrium. The initial calculations were useful in that they indicated the dependence of E_1 and E_2 (for a given configuration, density, temperature, and Xc) on R. It was found E_1 falls slightly in magnitude as R increases from 2.5 to 3 cell units, and that each of the 32 particles can be considered to have a roughly constant energy of interaction with the rest. Because of the oscillatory nature of the second term, the energy associated with a particular K particle varied considerably, and the sum of the 32 K contributions to give E_2 for the whole cell was small compared to E_1 for the cell. The R dependence of E_2 is in the factor $e^{-\pi^2/R^2}$, and so rapidly increases with R. Table 5 forms an attempted equilibrium configuration from 8 sets of four particles, temperature $10^{5\circ}$K, density 10^{18}e/cc, Xc=1.0 cell units, and gives the Energy terms in

(cell units)$^{-1}$ for a given configuration, where $E_3 = -\frac{R}{\sqrt{\pi}} \sum_k C_k^2$, and E_T is the total energy per cell.

TABLE 5

R	2	2.7	3.0	3.5
E_1	-164.915	-153.201	148.473	-140.877
E_2	+ .419	+ 1.565	+ 2.197	3.235
E_3	- 36.109	- 48.746	- 54.162	- 63.189
E_T	-200.605	-200.382	-200.438	- 200.831

Applying equation (2.13) to the figures above, it was found that optimum accuracy would be obtained by giving R a value of 2.7 (cell units)$^{-1}$. This value was subsequently tried, and compared with a very accurate calculation of the energy for the same configuration (i.e. R=2 X_c=2, $\Rightarrow D_o$=4, and E_1=-164.666, E_2=+.419, E_3=-36.109, E_T =-200.357) and for X_c=1.0 R=2.7 gave the minimum error. The accurate calculation for X_c=2.0 cell units however, was very time consuming, and it later became apparent that it would be better to sacrifice some accuracy for an increase in speed, as the number of iterations necessary for the system to come to equilibrium proved unexpectedly large. With this in mind X_c was reduced to .5 cell units, which now only includes the Debye shielding distance for temperatures below 5×10^{4}°K, and by going through the above analysis, a value

of R=3.5 was found to give the minimum error. This reduced the overall accuracy so that the energy now may contain an error of ±1 in the third figure, however the increase in speed, a factor of about five, amply justified the change. Also by cutting off at .5, this enabled quite a deal of simplification in the program, as now only the basic cell need be considered. Further discussion on the effects of the changes on the program and the results will be made in the relevant sections.

REFERENCES TO CHAPTER II

[1] Chapman,S., Cowling, T.G. "The Mathematical Theory of Non-uniform Gases". (Cambridge University Press: London) p5 (1953)

[2] Alder, B.J., and Wainright, A.W., Jour. Chem. Phys. 33 p.5 (1960)

[3] Born, M. and Huang, K., 'Dynamical Theory of Crystal Lattices' (Clarendon Press: Oxford) Appendix II (1954)

[4] Metropolis, V.M., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E., Jour. Chem. Phys. 21 p1087 (1953)

[5] Wood, W.W., and Parker, F.R., Jour. Chem. Phys. 27 p720 (1957)

III RESULTS, DISCUSSION AND CONCUSIONS

In this chapter the results will first be presented, with brief notes, in the order in which the calculations were done, and then followed by some more general notes on the graphs. The Debye-Hückel theory; work done by Broyles and Carley; Villars; and the Saha theory are each discussed in relation to the results, and then a general discussion of the results is given. Finally the conclusions and limitations of the method are presented.

3.1 Presentation of Results

3.1(a) Preliminary Results from 1620 on 4 particles. The first results were used to show that the program calculated the energy correctly, and that it operated at optimum speed and accuracy. Due to storage imitations only 4 particles could be considered on the I.B.M. 1620, but even so several features became quickly evident. The variation of the probability for a transition (PET) with energy; the dependence of the energy on the initial configuration; the increasing randomness as temperature was raised; the fact that pairing started to occur at about 10^{4}°K; were all apparent from the 4-particle calculations. Most effort was directed towards increasing the speed of calculation

while still maintaining the accuracy, and as discussed in the section on Accuracy, most of the parameters were fixed by these considerations.

3.1 (b) Results from IBM7090. Using the 4-particle results to start from a favorable configuration, the program was transferred to the larger computer, and 16 electrons and 16 protons considered in the unit cell. (Conversion factors for cell units of length and energy are found in Appendix I). The electrostatic energy of the 16 electrons and 16 protons in a particular configuration is termed the Cell Energy, and this is plotted against the number of iterations done. (An iteration is completed when all 32 particles in the cell have each been considered for possible movement.) When the energy was considered to have reached a stable value, the run was terminated, and graphs of the radial distribution functions for like and unlike particles were compiled from the equilibrium configurations, i.e. where the energy graph was relatively level. These graphs are presented collectively in the order that they were computed, and collective notes follow the graphs to explain more fully how they were compiled. The number density was constant for all runs, the number of protons equalling the number of electrons=10^{18}e/cc. The values of the other important parameters; δ_0, the maximum movement

a particle may undergo; X_c, the distance beyond which contributions from the first energy term are neglected; and the convergence parameter, R, are all included on the graphs.

Note 1: δ_o is in cell units, X_c in cell units, and R in (cell units)$^{-1}$, where the conversion factors for a cell unit of length are given in Appendix 1.

Note 2: Run 1 was later found to contain some small faults, thus truncation in $\delta_{\underline{x}}$ caused a consistent movement of particles in the negative direction, and an error in an input constant caused the first energy term to have about 3% error, but these errors should not greatly alter the graphs.

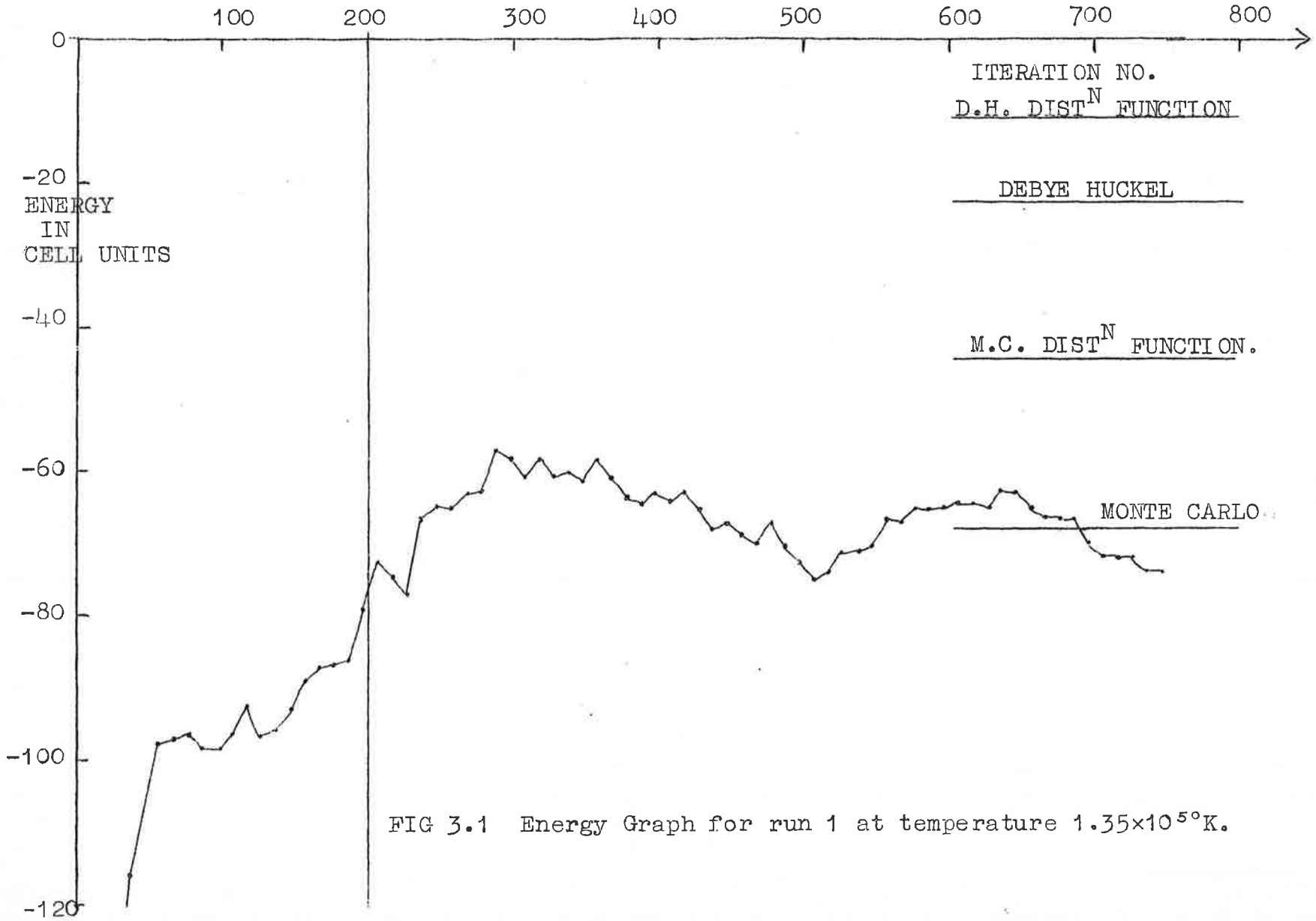


FIG 3.1 Energy Graph for run 1 at temperature 1.35×10^{5}°K.

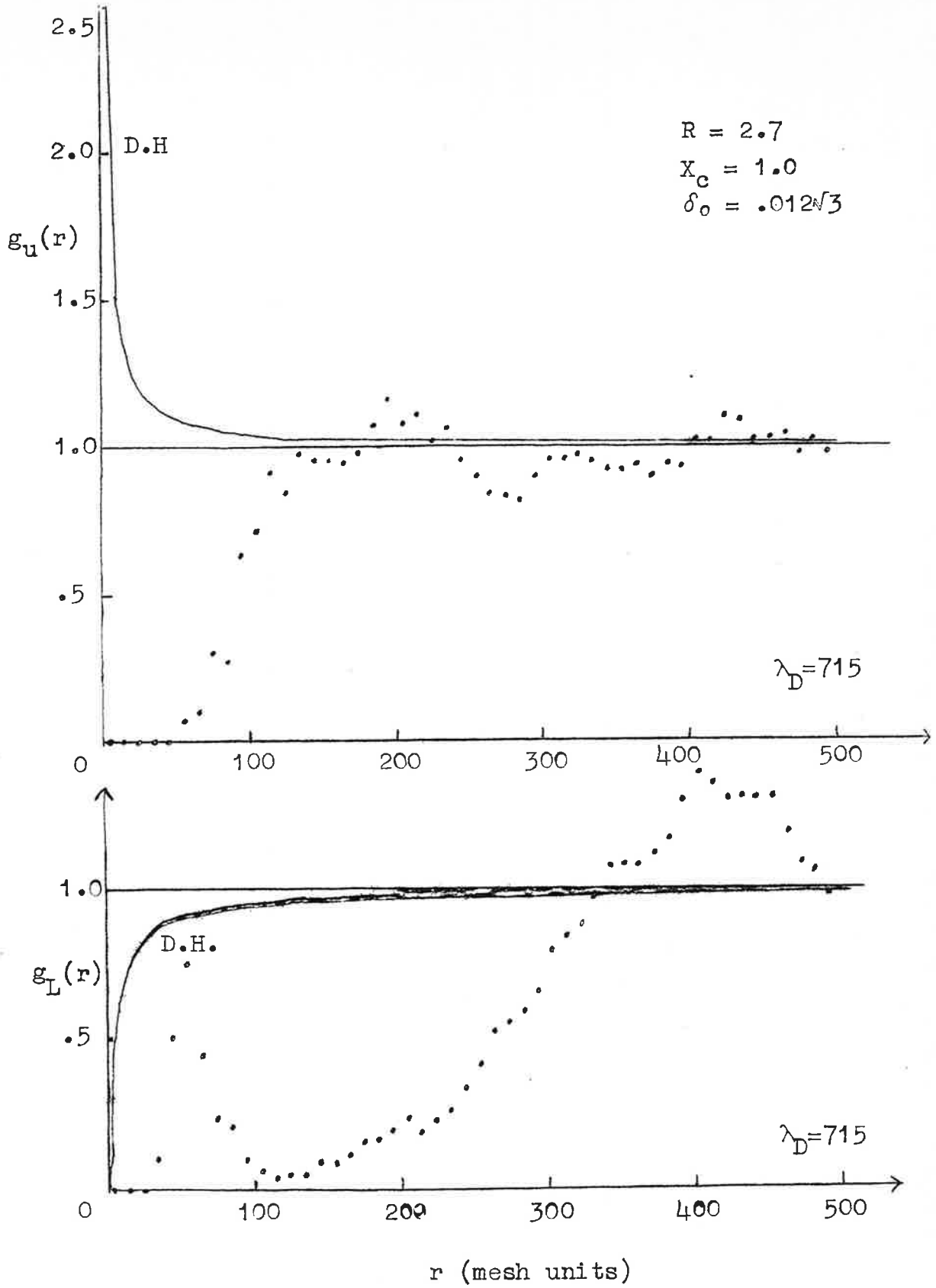


FIG 3.2 Radial Distribution Functions for run 1. ($T=1.35\times10^{5}$°K) taken from iterations 200 to 750.

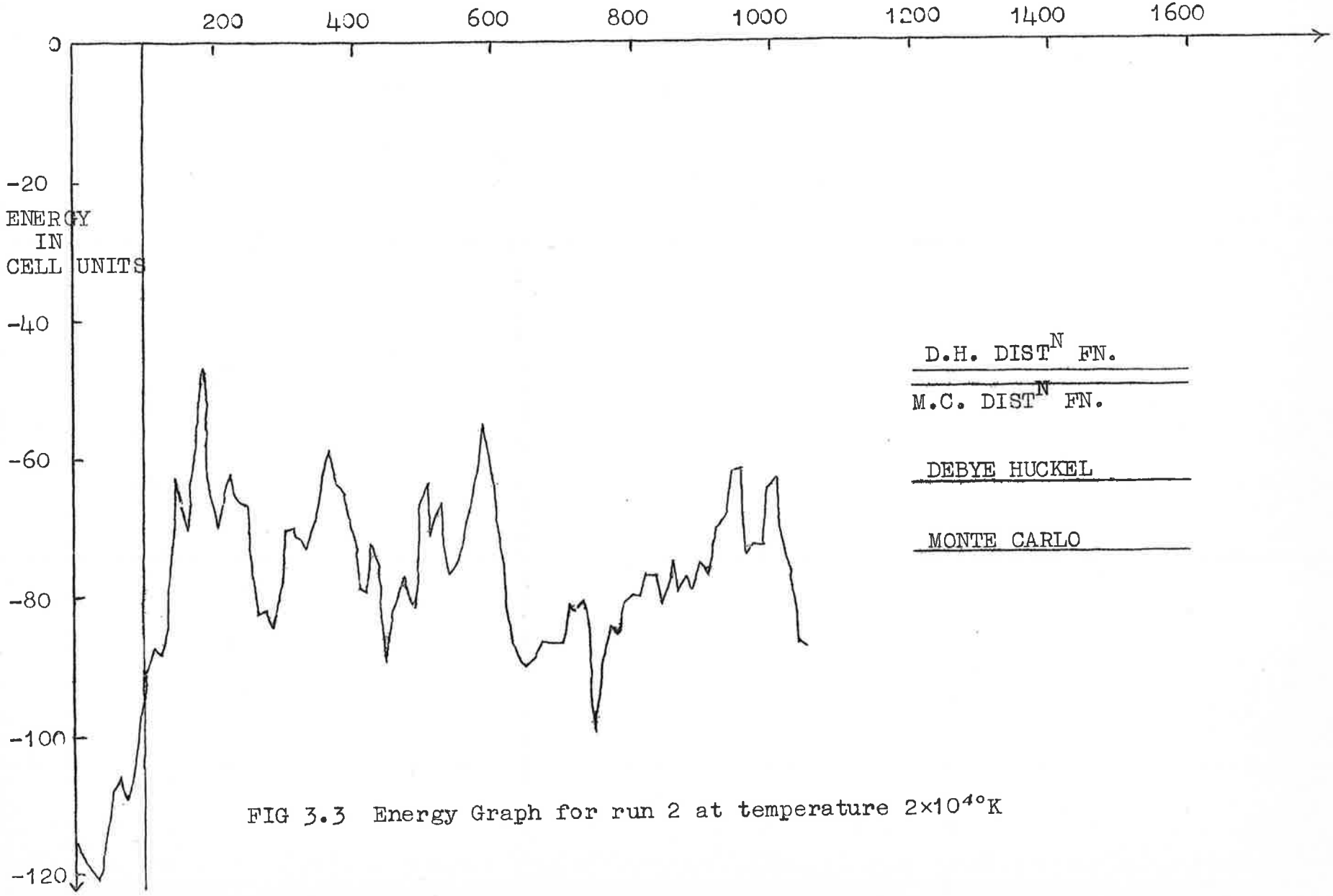


FIG 3.3 Energy Graph for run 2 at temperature 2×10^{4}°K

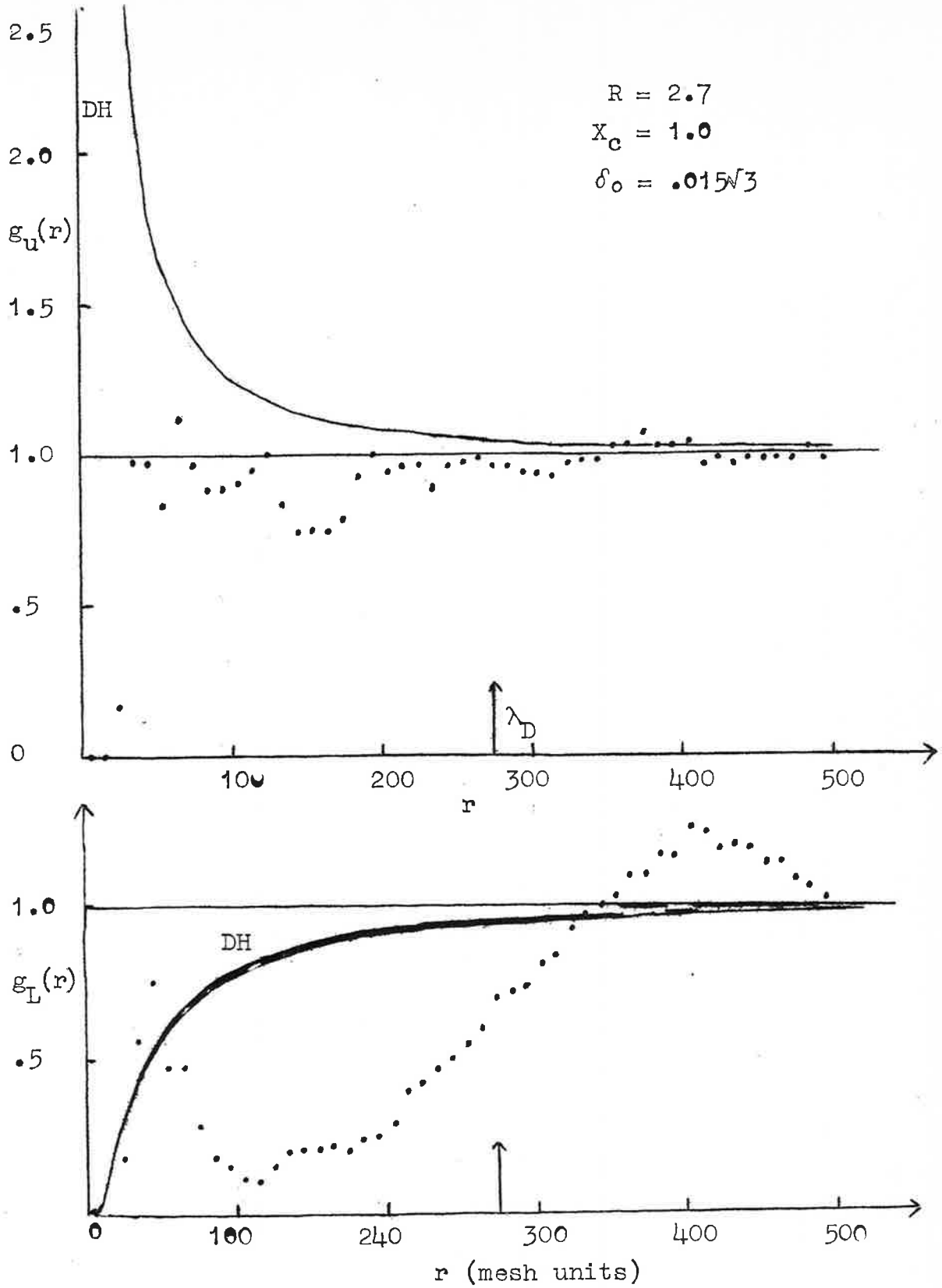


FIG 3.4 Radial Distribution Functions for run 2 ($T = 2\times10^{4}{}^{o}K$) taken from iterations 100 to 1050.

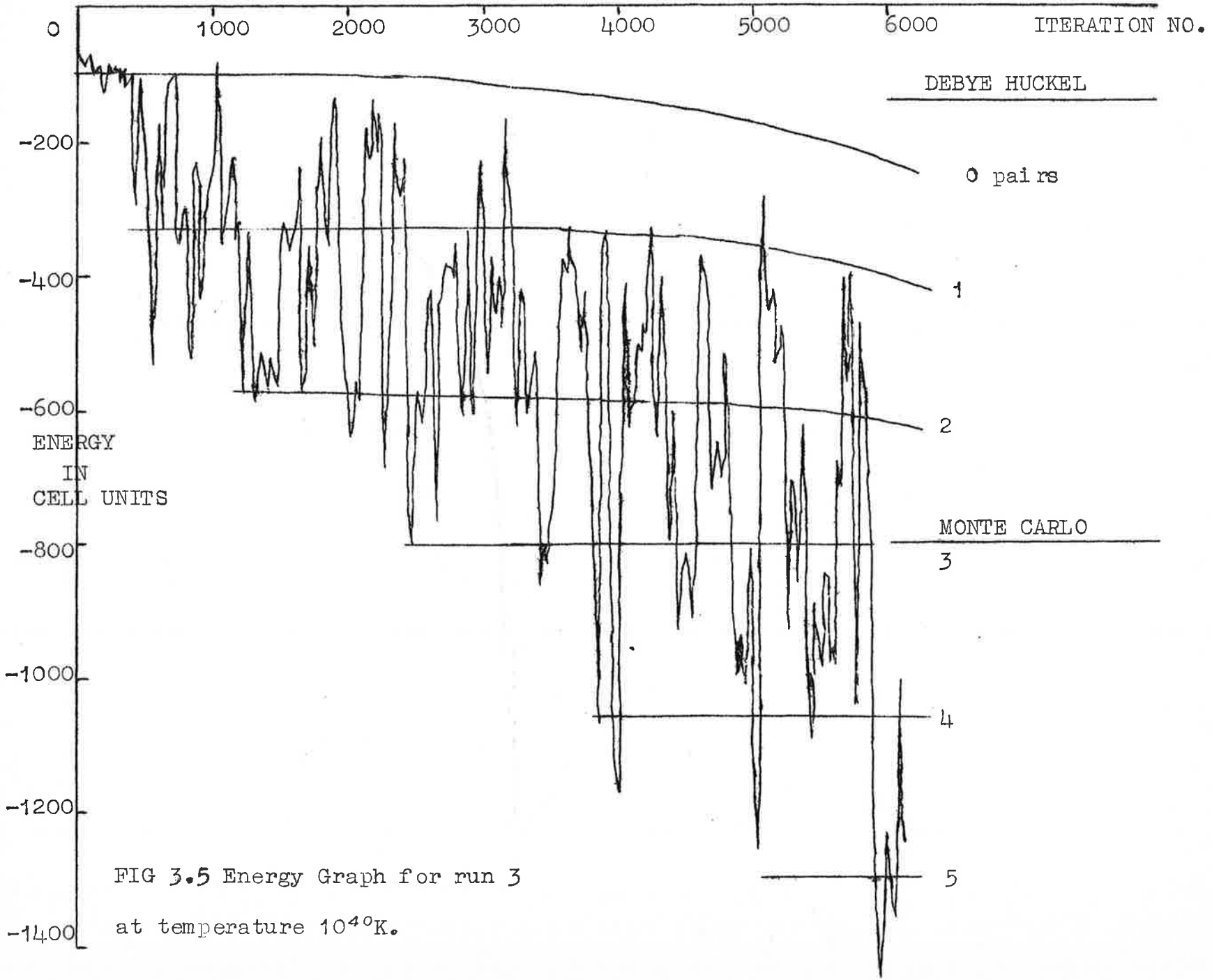


FIG 3.5 Energy Graph for run 3

at temperature 10^{4}°K.

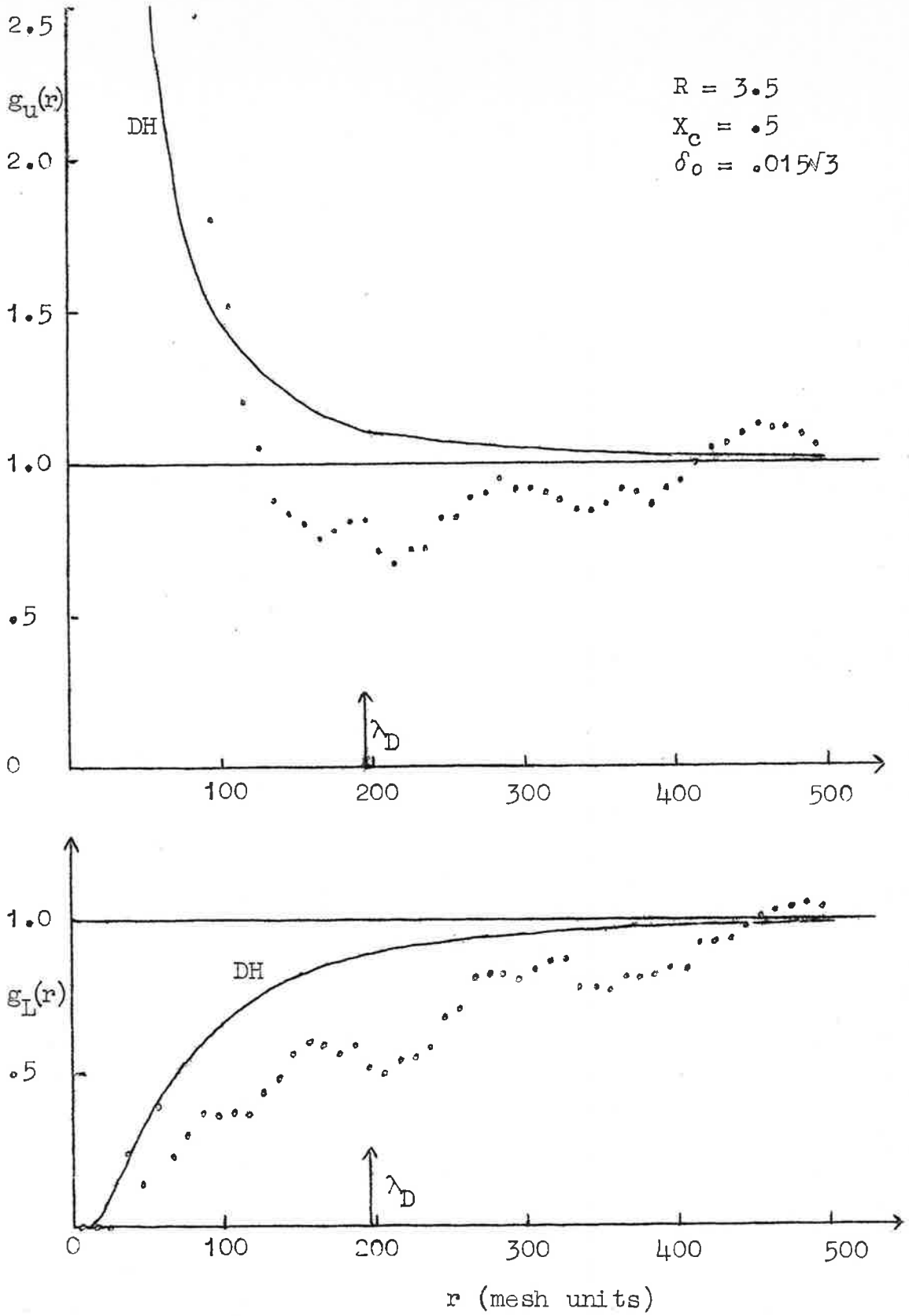


FIG 3.6 Radial Distribution Functions for run 3 ($T = 10^4$°K) taken from iterations 4000 to 6100.

3.1 (c) Notes on the graphs. The output from the program to obtain the radial distribution functions is in the form of 500 fixed point numbers for each distribution function, these numbers, denoted by ΔN_L (ΔN_U in the case of Unlike particles) give the number of pairing distances recorded in the interval r to r+Δr, where r goes from 0 to 500, proceeding in steps of $\Delta r=1$ mesh units. Then the estimate for the distribution function is obtained via equation (2.1). In order to graphically present the results, and indeed, to obtain anything like a smooth graph for g(r), it is necessary to sum the ΔN in steps of $\Delta r=10$, the ΔN being added manually and then averaged, while the corresponding r value is the average for the interval. It is to be observed that, the first 4 numbers of the output represent pairing, since, by the critereon for pairing from section 2.2, particles closer than A0=4.2 Mesh units are considered as paired. The radial distribution function is normalised, so that $g(r)\to 1$ as $r\to\infty$, but since the self-interaction distance is not counted, it follows that only 15 like inter-particle distances are recorded in the cell for one particle, so one may expect $\rho_o=32\times 15$ as the 32 particles are moved.

However if this is used in (2.1) to obtain the normalisation constant, then it leads to a large error when the energy/cell is calculated from the radial distribution function, as now the number of electrons (labelled a in equation (3.1)) is not equal to the number of protons (labelled b in equation (3.1)), and the cell is not electrically neutral, so the macroscopic energy components in the formulae

$$U_e = \Sigma_a \Sigma_b \frac{N_a N_b}{2} \sum_r (g(r)-1)\phi_{ab} 4\pi r^2 \Delta r \qquad (3.1)$$

do not cancel. This means ρ_o must be the same for like and unlike particles, and the radial distribution functions are obtained using normalisation factors with ρ_o=16.32 in both. It can be seen from the theory that $g(r) \to 1$ as $r \to \infty$, but unfortunately it was difficult to extend the

graphs further than 500 mesh units, without greatly increasing computational time. For comparison the radial distribution graphs include a graph of the Debye-Hückel radial distribution function, and the Debye-Shielding distance is marked by λ_D, to indicate the distance beyond which the charge of the reference particle should be neutralised.

To obtain the energy graphs from the computer output, which prints out the iteration number and the corresponding cell energy, the average is taken over 10 energy values, for runs 1 and 2, and this average is plotted against the mid-point iteration number. For the third, (much longer) run, it was necessary to average over 20 energy values, and even then the graph is highly erratic although before averaging it was even more so, large fluctuations occurring within a few iterations. Plotted on the right-hand side of the energy graphs, are various energy levels. The Monte Carlo level is the average energy of the calculation, and is obtained by averaging from the vertical line on the left hand side of the graph to the right hand edge. The extreme left hand side of the graph is neglected because the system has not approached anything like an equilibrium level, while the average is meant to give the equilibrium energy. The Debye-Hückel line

gives the cell energy from the Debye-Hückel theory discussed in 3.2. The M.C. $DIST^N.F^N$.line calculates the cell energy from the radial distribution graphs in the following manner. From equation (3.1) the electrostatic energy per cell is given by

$$U'_e = \tfrac{1}{2}\, 32.16 \sum_{r=o}^{500} (g_u(r)-1)\left(\frac{-1}{r}\right)4\pi r^2 \Delta r$$

$$+ \tfrac{1}{2}32.16 \sum_{r=o}^{500} (g_L(r)-1)\left(\frac{+1}{r}\right) 4\pi r^2 \Delta r \ ,$$

where the sum over r is divided into sections of $\Delta r=25$ mesh units, and hence U_e can be obtained manually from the graphs. The D.H. $DIST^N F^N$ level gives the corresponding Debye-Hückel energy for the range 0 to 500 mesh units, this being obtained from the D.H. distribution function, summing as above; and also by integration over the smaller limits a_o and b_o used in section 3.2. The graph, fig 3.5, does not have the $DIST^N F^N$ levels marked, because values obtained for these energy levels are very uncertain, due to the large values of g(r)-1 near the origin. Instead, this last graph, sloping horizontal lines are drawn which indicate the number of pairs existing (in one cell) at this energy level. The lines slope because of the critereon taken for pairing in section 2.2, so that although only one pair

may exist, there is an increasing number of links, or particles not far from pairing, as the calculation proceeds, and these links lower the energy level at which one pair is present. Before discussing the graphs and their implications in detail, the other theories will be briefly presented and discussed, so that later comparisons can be made between the results without forward referencing.

3.2 Debye-Hückel Theory

This theory, put forward by Debye and Hückel [1] in the 1920's and applied by them [2] to weak electrolytes, can also be applied to plasmas, i.e. fully ionized gases whose resultant macroscopic charge is zero. (It is also true that the technique used in this work could be used immediately to determine radial distribution functions in electrolytes, the only difference being that the dielectric constant must be included in the equations.) The theory is particularly attractive for its simplicity, as using the Boltzmann distribution law, Poisson's equation, and assuming that the interionic potential energy is small compared to the thermal energy, the following equation is obtained for the distribution function

$$g_{ab}(r) = \exp\left\{\frac{e_a e_b}{rkT} \quad \frac{1}{\exp(r/\lambda_D)}\right\} \qquad (3.2)$$

where $\lambda_D = \left(\frac{kT}{8\pi ne^2}\right)^{\frac{1}{2}}$, and is called the Debye Shielding distance, and

e_a, e_b are the charges on the particles of type a and b,

T is the absolute temperature,

n = no density of particles of type a = no density of particles of type b for a neutral binary plasma of singly ionised particles.

The theory implies that surrcunding a given ion there is a sheath of ions of opposite sign, and that outside a distance λ_D the sheath has neutralised the potential field of the ion; hence λ_D approximately measures the thickness of the ionic atmosphere, and also λ_D is often considered as the distance within which individual particle interactions are important, but beyond which a collective approach can be adopted. The graphs of $g_{DH}(r)$ in FIGS. 3.2, 3.4 and 3.6 emphasize the exponential form and lack of oscillation. The Debye-Hückel energy due to the electrostatic interactions is easily obtained from equation (3.1) by substituting $g_{DH}(r)$ and using the linear assumption or the superposition approximation. The compatability of the linear assumption and the superposition approximation is discussed in several books, see [3] and [4]. Fcr a symmetric plasma such that $n_a = n_b = n$ we obtain the energy/unit volume as

$$E_{D.H.} = \frac{ne^2}{\lambda_D}\left[e^{-b_o/\lambda_D} - e^{-a_o/\lambda_D}\right]$$

where b_o is the range of the distribution function

a_o is the distance of closest approach (AO)

The two levels plotted on the energy graphs are the total Debye-Hückel energy, where $a_o \simeq 0$ and $b_o \to \infty$; and the energy due to the radial distribution function, where $a_o = 0.0042$ and $b_o = 0.5$. The difference in the two levels emphasizes the contribution of the short range interactions, below 500 mesh units, to the energy, and from the graphs it can be seen that this increases as the temperature is lowered. In several calculations the Coulomb contribution of interactions within 0 to 500 mesh units was calculated, and this had a value very nearly equal to the M.C. $DIST^N$ F^N level.

The Debye-Hückel theory is known to be inaccurate when the density of the plasma is high, for then the interionic potential is appreciable compared to the thermal energy, and it has been found that concentrated electrolytic solutions show some properties of crystalline structure, in which case the radial distribution function should oscillate, in constrast to the D.H. form. To improve the D.H. theory for dense plasmas, and obtain improved values for the energy levels, Müller [5] and Gronwall, La Mer, and Sandved [6], expanded the $g_{DH}(r)$ to higher terms before using equation (3.1).

They noticed that for a symmetrical plasma all the even terms drop out, and hence such a plasma was more accurately described by the D.H. method, but even so as the density was increased they found appreciable discrepancies with the experimental energies.

In the cases in this work, the D.H. theory would not be expected to apply to the 10^{4}°K case, because of the neutral particles present, and also it is a bad approximation for the 2×10^{4}°K case, because here there is at least a 10% correction in the energy using formulae derived by [6]. However, the correction at 1.35×10^{5}°K is only 0.03% and although the plasma is fairly dense, now the thermal energy is becoming very large, and one would expect the Debye-Hückel theory to apply reasonably well.

3.3 Results of Broyles and Carley and Villars

A.A. Broyles, whose papers reviewing the methods for obtaining radial distribution functions in fluids were mentioned earlier, has recently, with D. Carley, extended the most successful of these, the Convolution hypernetted chain (CHNC), Perkus-Yevick (PY) and Broyles-Sahlin (BS) methods, to calculate the radial distribution functions for long range forces [7]. D. Carley [8] applies the techniques in detail to a classical electron gas

(i.e. electrons considered moving in a neutralising uniform background of positive charge), and presents distribution functions for various densities and temperatures, in terms of a parameter θ. For 10^{40}K and 10^{18}e/cc, his $\theta \sim 370 T°K/(Ne/cc)^{\frac{1}{3}}$ has a value of 3.7, and at this value he finds the radial distribution functions (for like particles) given by the P.Y and B.S methods agree well, but disagree with the D.H. and CHNC methods. This disagreement is enough to alter energy values considerably, but the form of the distribution function from all methods is very similar, and hence has not been drawn on the graphs, the D.H being deemed sufficient. He finds no oscillatory behaviour of g, but notes later that preliminary results for $\theta \sim 0.5$ begin to show an oscillatory nature for g, and points out that as θ becomes smaller solution of the integral equations becomes more difficult, the convergence being slower and less stable.

H.S. Green [9] put forward a set of integral equations for plasmas to arbitrary accuracy, and demonstrated that $g_{DH}(r)$ was an approximate solution of these equations. Recently Villars [10] has numerically solved these equations for metallic plasmas, and published the unlike radial distribution functions so obtained. For 10^{18}e/cc at 10^{40}K he obtains a curve very similar to the D.H curve,

but slightly displaced, this being significant when calculating some of the thermodynamic variables. During the numerical procedure he finds that it is necessary to use a D.H. tail to the radial distribution function, that g(r) must also be renormalised, and that if this is done then the results converge rapidly for temperatures of 10^{70}K and 10^{60}K, slowly for 10^{50}K, but diverge for 10^{40}K. However he still publishes the results!

3.4 Saha's Theory

In 1920 Saha [11] and [12] put forward a theory on the thermal ionisation of gases, which he considers from the view-point of chemical equilibria. Applying the law of Reaction Isobar, in which he replaces the entropy by an expression involving the Boltzmann distribution law, he finally obtains an equation for the degree of ionisation of the gas. This equation has been used extensively since he proposed it, and has proved reasonably successful. For a hydrogen gas density 10^{18}e/cc, he predicts that the degree of ionisation α (i.e. number density ions/ number density of atoms originally in the gas) is 5% at 10^{40}K, and approximately 90% at 2×10^{40}K. Referring to the basic cell of 32 atoms used in the Monte Carlo approach, this means that from Saha's theory one might expect one pair of particles at 2×10^{40}K, with the other

30 particles existing as ions, whereas at 10^{4^0}K one would expect only two ions and the other 30 atoms to be existing as 15 pairs. However, the Saha equation involves certain approximations, e.g. it neglects excited states, an ion being either in its lowest bound state, or in the continuum, and it also includes thermodynamic assumptions. As no other successful equations for the percentage ionisation of a gas have been developed, Saha's work provides the main basis for comparison on this point.

3.5 Discussion

From the theories just presented one might expect the following. At $1.35\times10^{5^0}$K the radial distribution function agreed well with $g_{DH}(r)$ and that at this temperature complete ionisation existed. But at $2\times10^{4^0}$K and 10^{4^0}K one might not be surprised if $g_{MC}(r)$ differed significantly from $g_{DH}(r)$, as at these temperatures ionisation is not complete, and the D.H. corrections are large.

Looking at Fig. 3.2 for $2.35\times10^{5^0}$K it can be seen that $g_{DH}(r)$ and $g_{MC}(r)$ differ enormously, and also the energy obtained by the M.C. method is quite different from the D.H. level. From later calculations it became apparent that the main cause of such a discrepancy is that, in spite of appearances, the system is not in its

equilibrium state by the end of the run because the system only comes to equilibrium very slowly. However, during the search for causes of the discrepancy, several interesting features came to light. The energy peaks are closely related to particle movements, and when unlike particles approach, a sudden downward peak appears, and conversely. The particles rarely come close together, being randomly dispersed throughout the volume of the box. In the run of 750 iterations with $\delta_0 = 12\sqrt{3}$ mesh units the particles, moving in a random walk, end up about 200 mesh units from where they started. Thus it seems that the computation has not run long enough for a close collision, and also has not run long enough to lose memory of the initial configuration. However, these results may be useful for obtaining information about irreversible processes, as the system is in the process of coming to equilibrium. In fact the system can be considered as particles in contact with a heat bath, and the particles observed for their interactions and movements. Then H.S. Green [13] has shown that the study of one particle throughout several collisions (which would be a longer run than the ones completed) is sufficient to determine the transport coefficients for the system, and so the results may be applied to this aspect. Alternatively they could be used to obtain the time dependence of the radial distribution

functions, and once this is known, then Wang-Chang and Uhlenbeck's [14] approach used to obtain the transport coefficients. Such an extension of these results may be attempted at a later date.

To rectify the errors in run 1, run 2 was made at a lower temperature, where the statistics should have more influence, δ_0 was increased to $15\sqrt{3}$ mesh units so that more of configuration space could be sampled, and the run was made longer, starting from a configuration obtained from the end of run 1, but with two pairs. The effect of increasing δ_0 can be noticed from FIG 3.3, where the energy now undergoes much larger fluctuations as the particles are moved; this may also be partly due to several weak links being set up between particles. The average energy level now obtained agrees reasonably closely with the D.H. level. However, looking at the radial distribution functions obtained (Fig 3.4) there is still considerable discrepancy between the MC and DH curves, though this discrepancy is now smaller. That the discrepancy has the same form may be due to starting from a similar initial configuration to run 1; note the two pairs parted quickly. The MC curve, especially of the unlike distribution function, is also rather ragged, which implies a longer run still is needed; and this conclusion

is strengthened by the fact that there have been no very close collisions. Because run 2 took considerable computing time and expense, it became evident that a longer run could not be considered unless the program was altered. This meant the accuracy of the energy calculation must be reduced, but by an acceptable amount.

Run 3 was made with such alterations, at a slightly lower temperature (which seemed of interest since pairing might occur), and started from the final configuration of run 2. From FIG 3.5 it can be seen that pairing, or very low energies, soon occurred, and there were large fluctuations as the pairs came together or jumped apart, this being a regular feature of the energy graph. In fact in the results which have not been averaged as in the energy graph, it can be seen that on some occasions several pairs suddenly split up, coming together some time later, and as the run continues, the number of pairs existing increases, allowing even larger fluctuations of energy. However, after 6000 iterations the energy graph does not seem to have settled into an equilibrium state, where one would expect occasional pairing and splitting up about a mean level, so the radial distribution functions will not be representative of an equilibrium state. Notice at the end of the run approximately three pairs are present on the average, whereas

Saha's theory suggested 15 pairs, also in run 2 there was no pairing when Saha suggested 1 pair. A brief run at 10^{30}K showed that at this temperature all particles remain paired, as Saha also predicts. This implies that perhaps ionization is slightly more sudden than Saha suggests, but such an implication needs to be verified by a longer run at 10^{40}K, so that equilibrium is reached, and similar runs at 9×10^{30}K and 1.5×10^{40}K. This may be attempted later if sufficient computing funds are available. FIG 3.6 shows the radial distribution functions for run 3, and although the graphs are still irregular, the general features are clearer. For unlike particles there is a peak indicating the definite preference for particles to pair, this being followed by a trough suggesting an absence of unlike particles, before the curve returns to something like the average value. Such a curve if fully verified, may indicate that some form of long range structure is being set up in the plasma. The structure is obviously not nearly as precise as for an ionic crystal, but it may well enhance the theory of Plasma Oscillations as postulated by Bohm and Pines [15], who have shown that collective excitations occur for an oscillation wavelength greater than the Debye shielding distance. As λ_D is small for a low-temperature plasma of high density, one would expect plasma oscillations

particularly in such plasmas. The Debye shielding length in FIG 3.6 occurs just before the minimum of the unlike radial distribution function, and so would fit in with the above suggestion.

The like distribution function is now becoming more like the D.H. curve, but has still not managed to fully fill-in the depression at about r=200, which has been present in all runs, and is most probably a feature of the initial configuration. Further, the depression in the unlike curve has become more marked with the longer runs, and enhances the concept that in equilibrium oscillation appears in the g(r).

3.6 Conclusions

The Monte Carlo method of calculating radial distribution functions in a plasma is a feasible approach if significant computing time is available. Compared to other methods it has the advantage of giving the percentage ionisation if the plasma is not fully ionised, and this is especially useful for dense plasmas. The results indicate that at least 10,000 iterations must be completed before the system can be considered near to its equilibrium state, and for a badly chosen starting configuration, the run would need to be considerably longer. This

being the case most emphasis in this work falls on the results from run 3 (at $10^{4}{}^{o}K$ with the number density of $10^{18}e/cc$), which indicates that the Saha equation for the degree of ionisation in a plasma is remarkably accurate, in that it predicts ionisation occurs between $10^{4}{}^{o}K$ and $2\times10^{4}{}^{o}K$. The distribution functions from run 3, imply, and the emphasis on this word must be stressed, that at in the region of this electron density oscillation appears in the radial distribution functions. However, for more conclusive results a longer run is needed, so the energy of the system can settle into an equilibrium pattern, and steady-state radial distribution functions obtained.

REFERENCES TO CHAPTER III

[1] Debye, P. and Hückel, E. Physik. Z. 24, p.185 (1923)

[2] Debye, P. and Hückel, E. Physik. Z. 31, p.797 (1930)

[3] Robinson, R.A. and Stokes, R.H. 'Electrolyte Solutions' (Butterworths Scientific Publications: London) p.74 (1955)

[4] Harned,H.S. and Owen,B.B. 'The Physical Chemistry of Electrolytic Solutions'. (Reinhold Publishing Corporation: New York) p.49 (1950)

[5] Muller, H. Physik. Z. 29, p.78 (1928)

[6] Gronwall, T.H.,La Mer,Y.K. and Sandved, K. Physik. Z. 29, p.358 (1928)

[7] Broyles, A.A., Sahlin, H.C., and Carley,D.D. Phys. Rev. Letters 10, p319 (1963)

[8] Carley, D.D. Phys. Rev. 131, p1406 (1963)

[9] Green, H.S. Nucl. Fusion 1, p69 (1961)

[10] Villars, D.S. Physics of Fluids 6, p745 (1963)

[11] Saha, M.N. Phil. Mag. 40, p 472-488 (1920)

[12] Saha, M.N. and Saha, N.K. 'A Treatise on Modern Physics'. (The Indian Press: Allahabad and Calcutta), p793 (1934)

[13] Green, H.S. 'Research Frontiers of Fluid Dynamics' (Edited by R.J., Saegar, to be published by John Wiley)

[14] Wang-Chang, C.S. and Uhlenbeck, P. (University of Michigan Engineering Report) (1952)

[15] Bohm, D. and Pines, D. Phys. Rev. 82, p625 (1951);85,p338 (1952)

IV COMPUTATION

4.1 Explanatory Notes

These explanatory notes refer closely to the listing given in section 4.2. It is a very full listing, incorporating the features of several programmes which could be run slightly more economically if run individually; for example a 'loader' program can be used to initially load the data, and after a few calculations put all the required information on tape, then a 'runner' program just reads off the tape, and continues the calculation. Brackets around the listing mean that this part of the program is not essential to the 'runner'. At the centre of the listing, an alternative listing is given on the right hand side, this second listing is much faster than the left hand side, but at the same time is less accurate, though speed is more important, as concluded in the results.

The listing will now be discussed in detail. Having dimensioned the particle coordinates, the pairing energy-distance matrix, and the radial index totals, the control variables used in mounting tapes and loading data are read in. (The specific nature of each variable is given in the symbol table, section 4.3). The required tapes are then loaded, and one of three methods can be used to initially specify the state of the system. First, 625 to 43, the coordinates of the particles and the pairing energy distance matrix can be read in by a DO loop, and

as the program goes through a few iterations, the pairing-energy matrix will assume its correct value. Secondly, 621 to 619, a specific record compiled in some previous run may be read off the tape and the calculation started from this point. Thirdly 619 to 460, a whole section of calculation may be stored on tape by first moving the calculation from another tape and restarting the calculation from the last record read. This elaborate procedure is necessary because on many computers the read and write heads can overlap, and it is only safe to read, OR write, on any one tape, otherwise there is a chance of overwriting some of the data which has just been read. This stage is only used if it is required to know detailed movements of the system, for it is expensive in time and tape space otherwise. Statement 400 provides a test to ensure the correct tape record has been read. From 409 the program decides whether to zero the pair distribution indices or whether to leave them, as the cumulative total for the complete run may be required. The provision for continuation runs is necessary, as a run of 1000 accurate iterations takes about five hours on a reasonably fast computer, such as the I.B.M. 7090. Finally before launching into the main calculation, a check is made, on both on-line and off-line printers, that the expected variables defining the starting configuration for this run, have been correctly stored; and several data cards are read in, overwriting those read from the

tape if necessary. During any one run these variables remain essentially the same.

The main DO loop controlling the number of major interations completed goes from 91 to 591. The top part of this nested DO, 91 to 33 zeroes the cell energy to be calculated, and restarts the calculation at the number one particle; while the latter part of the loop 84 to 591 determines whether output should be printed, written on tape, or ignored, it also ensures that if there is output that it is presented in the correct form. An IF (SENSE SWITCH,1) is also included to enable an operator to terminate the program at any time desired. A second major DO loop starts at 33 and goes to 84, and effectively sums over k in formulae (2.7). It considers the k^{th} particle in the main cell, and 33-30 decides whether it is a proton or an electron, (particles 1 to 16 are protons, 17 to 32 are electrons) and then zeroes the temporary (ETE) and permanent (EP) energies calculated for each particle in the loop. By the time the latter part of the loop 29 to 84 has been reached, ETE and EP associated with the k^{th} particle have been calculated, and the probability of the system going to the temporary configuration with the k particle in a new position is evaluated. A random number makes this decision. If EP is preferred, the permanent coordinates are kept, whereas if the decision is to move the

k^{th} particle to a new position with energy ETE, then the coordinates of this new position are retained, and the new pair distances and energies are put into the pairing energy-distance matrix to replace the previous ones. As the configuration has now been decided, the pairing distances between the k^{th} particle and all particles in the zero cell are added to the radial distribution functions for like ML(M) and unlike MU(M) pairing distances. This is done as follows, say an unlike particle is between 276 and 277 mesh units from the k^{th} particle under consideration, then a 1 is stored in the cumulative radial distance index labelled MU(277). Thus each time a k^{th} particle is considered, 16 units are stored in unlike indices, and only 15 in the like indices, since the k^{th} particle does not interact with itself. If a pairing distance is beyond 500 mesh units, it is stored in MU(500) or ML(500) depending on the type of particle. A third DO loop within the nest goes from 30 to 29. The latter part of this loop, 94 to 29 is used to decide whether the calculation just completed computed EP, in which case NDEL = -1 and the k^{th} particle must be moved by a random amount δ, or whether ETE was computed, in which case control passes out of this DO loop. The random numbers $\delta_{\underline{x}}$ are generated by the program itself 101 to 102, and a full discussion of the method is given in section 4.4. The random numbers

generated on (0,1) multiply an absolute displacement $A4=\delta_0$ and NA3 is subtracted (to ensure that the particles can move forward or backward) before being added to the original coordinates of the k^{th} particle. These new coordinates of the k^{th} particle are then used, by sending control back to 30, to calculate the temporary energy of the k^{th} particle.

The fourth DO loop 4 to 94, effectively carries out the sum over k' ($\equiv$KD) in formulae (2.7). If the loop is calculating the permanent energy of a particle k ($\equiv$ K), it only needs to add the correct interaction energies from the pairing energy-distance matrix to obtain EP, and it does this by considering the KD particles one at a time, picks out the interaction energy K-KD, then passes on to obtain energy K-KD+1 etc., and adding all the contributions gives EP. However, if the k^{th} particle has just been moved, and ETE is the energy to be calculated, then it will be necessary to calculate the new pairing energies and distances between K and the KD's for this new configuration. On considering a new KD particle, the first decision is whether the interacting particles K and KD are like (NQD = 1) or unlike (NQD = -1). The cell index NL is set to zero, as are the partial energy totals SUMHD and SUMEN. Due to the accuracy considerations of (2.3), where it was shown that the expansion over l, the cell number, need only be computed for $l_1 = -1,0,1$;

$l_2 = -1,0,1$; $l_3 = -1,0,1$; the coordinates of the interacting K and KD are first adjusted to obtain the most important interactions in this range. For example consider fig 4.1 with K at 100 in the zero cell, and KD at 900. [For computational purposes fixed point arithmetic is used, as it is much faster]. Then as the program procedes to calculate the interaction energy and distance between K and KD, it will first consider the interaction between K and KD in cell -1, the distance being 200. It next considers the zero cell, where K and KD are 800 units apart, and then cell +1 where K and KD are 1800 units apart. However this last interaction is not as important as the interaction between K and the KD in cell -2, where the separation is only 1200 units. Thus to take account of the most important interactions only, the following device is used. The distance with KD in the zero cell X(KD)-X(K)-2000=NXT is calculated, and if this NXT is less than -1500, 1000 is added. So in the above case NXT= 900-100-2000= -1200, and as this is greater than -1500, it

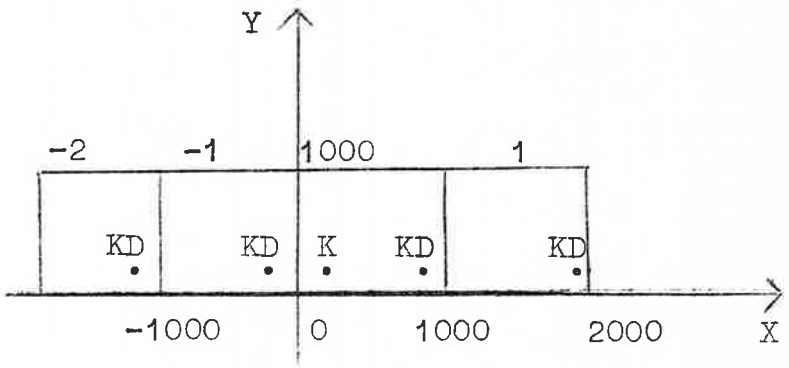


Fig 4.1 To illustrate the method of adjusting the particle coordinates for the X axis.

is retained as the first interaction distance. The next will be -200, the next 800, these being the three required. A similar procedure is adopted for the Y and Z coordinates, and the overall effect is to centralise the K particle in the lattice, so the optimum number of interactions are included. NOTE. The interaction energy between K and KD is the sum energy of several interactions, in fact the interactions between K **with** adjusted coordinates and the KD in the -1 cell, the KD in the 0 cell, and the KD in the +1 cell, and similarly for cells in the Y and Z directions; while the pairing distance between K and KD is the distance between the adjusted K particle and the nearest KD particle only, which is in the zero cell.

The summation over the cells is done by looping from statement 12 to 35. The alternative program given for this looping is much faster because only one cell need be considered, which as well as excluding many calculations of interaction energy, also removes the use of cell indices, however, as noted in 2.3, it is much less accurate. As it is similar to the more complete listing, only the latter will be discussed. The cell index NL is started from 0, which represents $l_1=l_2=l_3=-1$ and has a value 1 when $l_1=0, l_2=l_1=-1$ etc.; the zero cell, $l_1=l_2=l_3=0$, occurs when NL=13, and the final cell, $l_1=l_2=l_3=1$ when NL=26. The second energy term

EN of formula (2.7) needs to be evaluated only in the zero cell, and is computed by 20 to 37 if this is the case, unless K=KD when the evaluation can be written in the simpler form of 22. The first energy term, however, may receive contributions from other cells. To find the contribution from interaction between K and the KD in the first cell, the square of the distance between K and that KD is calculated (37), if this is greater than A2 the contribution will be negligible (See 2.3 for determination of A2) and control passed immediately to the next cell via 14; but if the square of the distance is less than A2 there will be a significant contribution HD as calculated in 21. (Note that calculation of HD uses a subroutine, ERRF(D) and this will be discussed in 4.4). The HD contributions are added for the 27 cells to obtain SUM HD, which is the first term part of the interaction energy between K and the KD. The small section 26 to 24, only used for the zero cell, stores the pairing distance between K and the nearest KD particle, which is in the zero cell after the coordinate adjustment made previously; it secondly ensures that if this distance becomes smaller than the potential well cut-off A0 (see 2.2d) then the distance is replaced by A0 before the energy contribution is calculated. After the energy calculation the cell index increases, and a COMPUTED GO TO statement sends

control to the next cell, and the next energy contribution is calculated. After all the contributions from the KD's in each cell have been summed, the interaction energy ET(KD) between the K^{th} and KD^{th} particle is obtained (38) by adding the first and second terms together, and putting them in the correct units. [Later this ET(KD) is stored via 36 in the pairing energy-distance matrix PE1(K,KD) if the configuration becomes accepted]. Control then passes to 94 and the next KD particle and images interacting with the K^{th} particle are considered. Thus as the program proceeds out through the nested loops, it decides on the position of the K^{th} particle, then sums over the K particles until all the K particles in the unit cell have been considered, and the sum given in equation (2.7) is completed at 156, and is printed out. At certain stages, when the number of iterations equals NUMB5, a full print out of the pairing-energy distance matrix and the pair distribution cumulative totals is obtained. However, the time for one accurate iteration is slow, about one third of a minute, thus output must be printed about every 25 iterations, so in case of machine failure not too much calculation is lost. Finally if the IF SENSE SWITCH is on, or if the number of iterations (NUMB1) is greater than NUMB2, control goes out of the program, and it then writes the final values of nearly all variables on tape and printer,

and finishes. Adequate storage facilities are available in the 7090 computer.

A few comments are probably in order concerning the data deck, a typical example of which is given directly after the listing. The first card contains the numbers of the tape units to be used, and a control variable deciding the number of tapes to be used, while the second card contains 8 fixed point numbers all concerned with the input data to correctly start the run. Then 32 cards are listed, giving the coordinates of 32 particles, which are only read in at the very commencement of a run when no tape input is available, otherwise they can be left out. Just before the main computation starts, other data cards, usually the third, fourth fifth and sixth, are read in. The third contains 9 fixed point numbers mainly concerned with the output of the data, while the fourth and fifth contain floating point constants associated with temperature, density, conversion factors, cut-off, and the initial random number. The sixth card contains 3 floating point variables introduced when the alternative fast program was made up, and are just functions of variables used in the slower listing. During a particular run, these last three cards remain the same, except for the random number, which is read in even on continuation runs, to ensure that the random sequence continues from the point

left off. Hence there are 6 main input cards, 3 of which alter every continuation run, and three which remain constant except for the random number. At the commencement of a new run the whole 6 cards are subject to change , in this case a careful watch must also be kept on DIMENSION statements, and the dimension of some of the output, as variation in these is impossible without recompiling. Finally Fig 4.2 gives the flow chart appropriate to the listing.

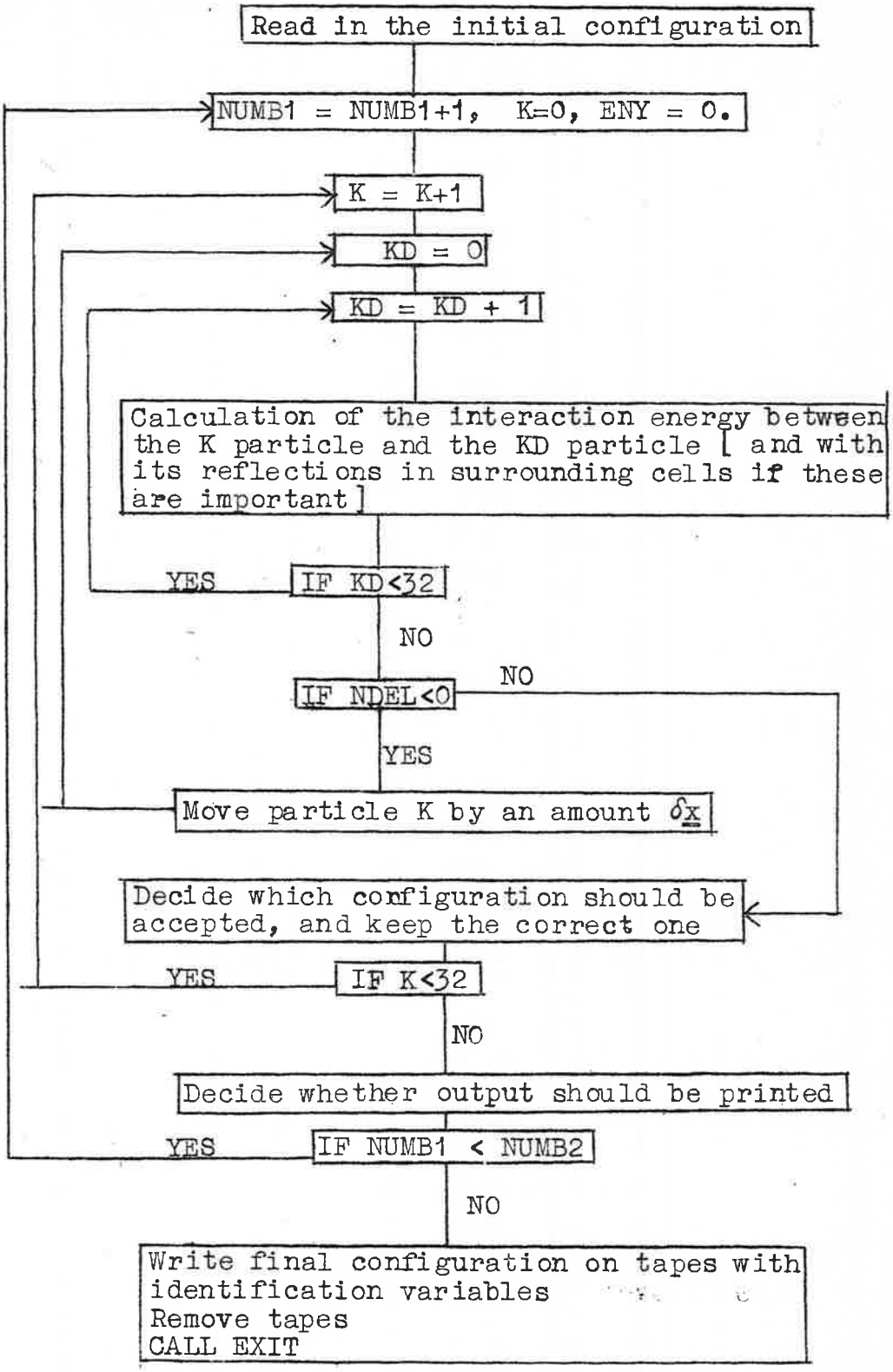


FIG 4.2 FLOW CHART

4.12 Program listing


```
*      IE5 616        UNI OF ADEL BARKER MATHS PHYSICS        C 98TX907     PX
*      XEQ
*      EXEC TIME 7 MIN
*      LINES    (3000)
*      CARDS   COLUMN
       DIMENSIONNX(32),NY(32),NZ(32),PE1(32,32),TD(32),ET(32)
       DIMENSIONML(500),MU(500)
       READINPUTTAPE2,221,JTAPE,KTAPE,LTAPE,MTAPE,NUM12
       READ INPUTTAPE2,221,NUMB8,NUMB6,NUMB9,NA2,NUMB1,NUMB4,NUMB7,NA4
  221  FORMAT(14I5)
       PRINT563,JTAPE
  563  FORMAT(1X31H LOAD UNI TAPE BARKER  NO.8 ON,I3,29HAND PRESS START
      1WHEN READY /////)
       PAUSE
       REWIND JTAPE
       CALL DNSHI(JTAPE)
       IF(NUM12)564,564,565
  564  PRINT562,KTAPE
  562  FORMAT(1X31H LOAD UNI TAPE NILSSON NO.3 ON,I3,29HAND PRESS START
      1WHEN READY /////)
       PAUSE
       REWIND KTAPE
       CALL DNSHI(KTAPE)
  565  IF(NUMB8)625,619,621
  625  DO17I=1,NA2
   17  READINPUTTAPE2,44,NX(I),NY(I),NZ(I)
   44  FORMAT(3I4)
       DO43K=1,NA2
       DO43KD=1,NA2
       IF(K-KD)629,629,630
  629  PE1(KD,K)=1.0
       PE1(K,KD)=0.1
       GOTO43
  630  PE1(K,KD)=1.0
       PE1(KD,K)=0.1
   43  CONTINUE
       GOTO400
  621  READTAPELTAPE,NUMB1
       IF(NUMB1-NUMB6)621,622,621
  622  BACKSPACELTAPE
       READ TAPELTAPE,NUMB1,((PE1(K,KD),K=1,32),KD=1,32),(NX(I),I=1,32),
      1(NY(I),I=1,32),(NZ(I),I=1,32),(MU(M),M=1,500),(ML(M),M=1,500),
      2NA1,NA2,NA3,PDL,A1,A2,A4,A0,BETA,A3,B4,B6,B7,R,A8,ENY
       GOTO400
  619  DO460NUMT=1,NUMB9
       READ TAPELTAPE,NUMB1,((PE1(K,KD),K=1,32),KD=1,32),(NX(I),I=1,32),
      1(NY(I),I=1,32),(NZ(I),I=1,32),(MU(M),M=1,500),(ML(M),M=1,500),
      2NA1,NA2,NA3,PDL,A1,A2,A4,A0,BETA,A3,B4,B6,B7,R,A8,ENY
```

```
460 WRITETAPEMTAPE,NUMB1,((PE1(K,KD),K=1,32),KD=1,32),(NX(I),I=1,32),
   1(NY(I),I=1,32),(NZ(I),I=1,32),(MU(M),M=1,500),(ML(M),M=1,500),
   2NA1,NA2,NA3,PDL,A1,A2,A4,A0,BETA,A3,B4,B6,B7,R,A8,ENY
400 IF(NUMB1-NUMB4)401,409,401
409 IF(NUMB7)191,291,291
191 DO100M=1,NA4
    ML(M)=0
100 MU(M)=0
291 PRINT584,NUMB1,NX(30),NY(30),NZ(30),MU(25),ML(495),
   1PE1(1,30),PE1(31,1),ENY
    WRITEOUTPUTTAPE3,584,NUMB1,NX(30),NY(30),NZ(30),MU(25),ML(495),
   1PE1(1,30),PE1(31,1),ENY
584 FORMAT(6I5,3F14.8)
    READINPUTTAPE2,221,NUMB1,NUMB2,NUMB3,NUMB5,NUM10,NA1,NA3,MUC,NRN
627 READINPUTTAPE2,77,A0,BETA,A3,B4,B6,B7
 77 FORMAT(6F12.8)
    READINPUTTAPE2,2211,A1,PDL,A2,A4,A8,R
2211 FORMAT(4F12.3,2F12.9)
    READINPUTTAPE2,291,SA0,ENC,TNA5
291 FORMAT(3F14.9)
    ENT=(3.0*(1.0+B6))+B7
 91 ENY=0.
    NUMB1=NUMB1+1
    K=0
 33 K=K+1
    IF(K-NA1)50,50,51
 50 NQ=1
    GOTO55
 51 NQ=-1
 55 ETE=0.
    EP=0.
    NDEL=-1
 30 KD=0
    TSMEN=0.
    TSMHD=0.
  4 KD=KD+1
    HD=0.
    IF(NDEL)48,46,46
 46 IF(KD-NA1)52,52,53
 52 NQD=NQ
    GOTO54
 53 NQD=-NQ
 54 NL=0
    SUMHD=0.
    SUMEN=0.
    NXT=NX(KD)-NX(K)-2000
 18 IF(NXT+1500)5,6,6
  5 NXT=NXT+1000
    GOTO18
  6 NYT=NY(KD)-NY(K)-2000
```

```
  7 IF(NYT+1500)8,9,9
  8 NYT=NYT+1000
    GOTO7
  9 NZT=NZ(KD)-NZ(K)-2000
 10 IF(NZT+1500)11,12,12
 11 NZT=NZT+1000
    GOTO10
 12 XT=NXT
    YT=NYT
    ZT=NZT
    IF(NL-13)23,25,23
 23 EN=0.
    GOTO37
 25 IF(K-KD)20,22,20
 20 D=COSF(B4*XT)
    E=COSF(B4*YT)
    F=COSF(B4*ZT)
    B=E+F
    C=E*F
    EN=D+B+(D*B+C)*B6+D*C*B7
 37 DS=XT*XT+YT*YT+ZT*ZT
    IF(DS-A2)13,114,114
 13 D=SQRTF(DS)
    IF(NL-13)21,26,21
 26 DT=D+TNA5
    IF(DT-PDL)74,74,58
 58 DT=500.0
 74 IF(NQD)71,71,72
 71 TD(KD)=-DT
    GOTO73
 72 TD(KD)=DT
 73 IF(D-A0)24,24,21
 24 D=A0
 21 D=D/A1
    HD=(1.0-ERRF(D))/D
    GOTO60
 22 EN=ENT
    HD=0.
    TD(KD)=1.0
114 HD=0.
 60 IF(NQD)16,15,15
 15 SUMHD=SUMHD+HD
    SUMEN=SUMEN+EN
    GOTO14
 16 SUMHD=SUMHD-HD
    SUMEN=SUMEN-EN
 14 NL=NL+1
    GOTO(1,1,2,1,1,2,1,1,3,1,1,2,1,1,2,1,1,3,1,1,2,1,1,2,1,1,35),NL
  1 NXT=NXT+1000
    GOTO12
```

```
12 XT=NXT
   YT=NYT
   ZT=NZT
   IF(K-KD)20,22,20
22 EN=ENT
   TD(KD)=1.0
   GOTO35
20 D=COSF(B4*XT)
   E=COSF(B4*YT)
   F=COSF(B4*ZT)
   B=E+F
   C=E*F
   EN=D+B+(D*B+C)*B6+D*C*B7
   IF(NQD)121,137,137
121 EN=-EN
137 DS=XT*XT+YT*YT+ZT*ZT
   D=SQRTF(DS)
   DT=D+1.0
   IF(DT-PDL)74,74,58
58 DT=500.0
74 IF(NQD)71,71,72
71 TD(KD)=-DT
   GOTO73
72 TD(KD)=DT
73 IF(DS-A2)25,35,35
25 IF(DS-SA0)24,24,21
24 D=A0
21 D=D/A1
   HD=(1.0-ERRF(D))/D
   IF(NQD)16,35,35
16 HD=-HD
```

```
  2 NXT=NXT-2000
    NYT=NYT+1000
    GOTO12
  3 NXT=NXT-2000
    NYT=NYT-2000
    NZT=NZT+1000
    GOTO12
 35 ET(KD)=A8*SUMHD+A3*SUMEN
    ETE=ETE+ET(KD)
    TSMEN=TSMEN+EN*A3
    TSMHD=TSMHD+HD*A8
    GOTO94
 48 IF(K-KD)62,62,63
 62 EP=EP+PE1(K,KD)
    GOTO94
 63 EP=EP+PE1(KD,K)
 94 IF(KD-NA2)4,34,34
 34 IF(NDEL)28,28,29
 28 NXP=NX(K)
    NYP=NY(K)
    NZP=NZ(K)
    NUM=1
101 R=(129.*R)+.7886751346
    NR=R
    RN=NR
    R=R-RN
    GOTO(102,103,104,105),NUM
102 NXD=A4*R
    NUM=2
    GOTO101
103 NYD=A4*R
    NUM=3
    GOTO101
104 NZD=A4*R
    NX(K)=NXP+NXD-NA3
    NY(K)=NYP+NYD-NA3
    NZ(K)=NZP+NZD-NA3
    NDEL=NDEL+2
    GOTO30
 29 PET=1.0/(1.0+EXPF(BETA*(ETE-EP)))
    NUM=4
    GOTO101
105 IF(R-PET)31,31,32
 32 ENY=ENY+EP-ENC
    NX(K)=NXP
    NY(K)=NYP
    NZ(K)=NZP
    GOTO85
 31 ENY=ENY+ETE-ENC
    DO40KD=1,NA2
```

101 NRN = (129 * NRN) + 103373
RTEM = NRN
R = RTEM / 131071.

```
      IF(K-KD)36,38,38
   38 D=TD(KD)
      E=ET(KD)
      GOTO39
   36 D=ET(KD)
      E=TD(KD)
   39 PE1(K,KD)=D
   40 PE1(KD,K)=E
   85 DO84KD=1,NA2
      IF(KD-K)83,84,86
   83 NQ=K
      NQD=KD
      GOTO87
   86 NQ=KD
      NQD=K
   87 M=(PE1(NQ,NQD)/TNA5)
      IF(M)81,81,82
   81 M=-M
      MU(M)=MU(M)+1
      GOTO84
   82 ML(M)=ML(M)+1
   84 CONTINUE
      IF(NUMB1-NUMB5)513,512,513
  512 WRITEOUTPUTTAPE3,5821,NUMB1,K,NXD,NYD,NZD,ETE,EP,PET,R,ENY,TSMEN,
     1TSMHD
 5821 FORMAT(1X5I4,7F14.9)
  513 IF(K-NA2)33,156,156
  156 PRINT587,NUMB1,ENY
  587 FORMAT(I5,F16.8)
      IF(NUMB1-NUMB5)402,403,402
  403 NUMB5=NUMB5+NUMB3
      WRITEOUTPUTTAPE3,582,NUMB1,ENY
  582 FORMAT(1H1/1XI5,F16.8)
      KJ=1
      KF=8
  160 WRITE OUTPUT TAPE 3,154,((PE1(K,I),K=KJ,KF),I=1,32)
  154 FORMAT(1X8F14.8)
      IF(KF-32)155,159,159
  155 KJ=KJ+8
      KF=KF+8
      WRITEOUTPUTTAPE3,1155,NUMB1
 1155 FORMAT(1H1/1XI5)
      GOTO160
  159 WRITEOUTPUTTAPE3,1159,(I,I=1,32)
 1159 FORMAT(1H1/1X,32I4)
      WRITEOUTPUTTAPE3,153,(NX(I),I=1,32),(NY(I),I=1,32),(NZ(I),I=1,32)
  153 FORMAT(1X32I4)
      WRITE OUTPUT TAPE 3,158,(MU(M),M=1,MUC)
      WRITE OUTPUT TAPE 3,158,(ML(M),M=1,MUC)
  158 FORMAT(1X25I5)
```

```
  402 IF(NUMB1-NUMB2)591,411,411
  591 IF(NUM10)408,408,91
  408 WRITETAPEMTAPE,NUMB1,((PE1(K,KD),K=1,32),KD=1,32),(NX(I),I=1,32),
     1(NY(I),I=1,32),(NZ(I),I=1,32),(MU(M),M=1,500),(ML(M),M=1,500),
     2NA1,NA2,NA3,PDL,A1,A2,A4,A0,BETA,A3,B4,B6,B7,R,A8,ENY
      GOTO91
  411 WRITETAPEMTAPE,NUMB1,((PE1(K,KD),K=1,32),KD=1,32),(NX(I),I=1,32),
     1(NY(I),I=1,32),(NZ(I),I=1,32),(MU(M),M=1,500),(ML(M),M=1,500),
     2NA1,NA2,NA3,PDL,A1,A2,A4,A0,BETA,A3,B4,B6,B7,R,A8,ENY
      WRITEOUTPUTTAPE3,1157,JTAPE,KTAPE,LTAPE,MTAPE
 1157 FORMAT(1X10H BARKER ON I3,11H NILSSON ON I3,5H READ I3,6H WRITEI3)
  401 WRITEOUTPUTTAPE3,150,NUMB8,NUMB6,NUMB9,NA2,NUMB1,NUMB4,NUMB7,NA4
  150 FORMAT(1X14HIDENTIFICATION/1X14I5)
      WRITEOUTPUTTAPE3,2222,NUMB1,NUMB2,NUMB3,NUMB5,NUM10,NA1,NA3,NA5
 2222 FORMAT(1X14I5)
      WRITEOUTPUTTAPE3,151,A0,BETA,A3,B4,B6,B7
  151 FORMAT(1X10HINPUT DATA/1X7F14.9)
      WRITEOUTPUTTAPE3,152,A1,PDL,A2,A4,A8,R
  152 FORMAT(1X4F12.3,2F12.9)
      WRITEOUTPUTTAPE3,1291,SA0,ENC,TNA5
 1291 FORMAT(1X3F14.9)
      REWIND JTAPE
      PRINT517,JTAPE
  517 FORMAT(1X34H REMOVE UNI TAPE BARKER  NO.8 FROM,I3,  ///)
      REWIND KTAPE
      PRINT 516,KTAPE
  516 FORMAT(1X34H REMOVE UNI TAPE NILSSONNO.3 FROM,I3,  ///)
      CALL EXIT
      END
)5 X'G  7 '(*XP   *PP  '                                                    A
*J Z'£7.,,W 7 99'(4 '9 '5 '7 =7 99'(4 79 '5 '9'(4 '9 '7'(7 '9'(G474 'G4'A
*J7'(X****N 5'(7'(7'(64'9'(-4'4 7G4'7'(7'(9'(-774 74 77'(E'(4 7='(G47'  A
*J7()PP)0   0 '07'7 86 79'(7'(7 '5494 75 '(((XB'X('X(((((X)(P  P  *Z( 0 A
```

```
*      DATA
    17    17    17    17    +1
    -2     0     0    32     0     0    -2   500
  496  -90  123
  732  237  382
  872  -40  221
 1110  281  425
  331  600  184
  590  896  347
  801  648  -46
  992  940  323
  407  205  591
  589  364  741
  849   99  618
 1099  380  805
  445  564  672
  809  848  970
  891  714  683
 1136  907  973
  398  226  214
  650  526  412
  884  250   64
 1182  485  295
  383  770  171
  626  882  378
  826  594  159
  984  833  281
  219  131  620
  532  357  921
  691  146  715
  881 3291259
  305  553  944
  562 8021052
  885  383  792
 1137  622  788
     0     9     2     1    +2    16    15     1  50027932
   4.20          .066310      .08220128    .00628318    .2582429     .08891919
      370.370      500.000 1000000.000       31.000 1.350000000   .235417862
   17.64           1.974663000   1.0
```

4.3 Symbol Table

A0 Input Data. This potential well cut-off (see 2.2(d)) depends on the density being considered, and for 10^{18}e/cc it has a value of .0042 cell units (see Appendix 1). Used in 73 it ensures two particles never come closer than A0, at least for energy computation purposes.

A1 Input Data. Converts the distance between K and KD in mesh units to cell units and at the same time multiplies by R (in inverse cell units). Hence it has in 21 a value 100./R cell units.

A2 Input Data. Controls the distance squared at which the contribution of the first term to the energy may be neglected via the IF statement near 37. Its magnitude is a critical factor in the speed and accuracy of the calculation as discussed in 2.3(b), where A2 = X_c^2.

A3 Input Data. Multiplies the second energy term in 35 by a factor $e^{-\pi^2/R^2}/\Pi$ to give it its correct absolute value as required from equation (2.7.)

A4 Input Data. Multiplies the random numbers generated on (0,1) to obtain the random displacement $\delta_{\underline{x}}$, with components given by 102,103,104, that the particle is to undergo. It's choice is discussed in detail in 2.2(c), where $3(\frac{A4-1}{2})^2 = \delta_o^2$

A8 Input Data. Multiplies the first energy term in 35 by R/2 to give it its correct absolute value as required from equation (2.7.)

B Defined near 20, this variable enables the second energy term to be evaluated quickly.

BETA Input Data. Reciprocal of the product of the Boltzmann constant (converted to cell units in Appendix 1) and the temperature in degrees Kelvin. It is the only input data that varies directly with temperature, and is used in the probability distribution function 29, the form of which was discussed in 2.2(b).

B4 Input Data. This converts XT into cell units and at the same time multiplies by 2π. Used in 20 it has a value $\frac{2\pi}{1000}$.

B6 Input Data. Used near 37, B6 = $\frac{e^{-2\pi^2/R^2}}{\pi \quad A3}$, and enables the second energy term to be computed quickly.

B7 Input Data. Used as above $B7 = \frac{4e^{-3\pi^2/R^2}}{3\pi \quad A3}$

C Defined above 37 as E*F for quick computation of second energy term.

D (a) Defined in 20 as COSF(B4*XT) for quick computation of second energy term.

(b) Defined in 13 as SQRTF(DS). This gives the distance between the interacting K and KD particles in mesh units.

(c) May be defined in 24 as D=A0 if particles come too close together.

(d) Later redefined in 21 as D=D/A1, this puts D in a form such that it can be used directly to evaluate the first energy term contribution.

DS Defined in 37 by DS = XT*XT+YT*YT+ZT*ZT, this is the distance squared between particles K and KD in mesh units squared.

DT Defined in 26 by DT=D+1.0, this adds one mesh unit to the inter-particle distance in mesh units, to allow for truncation later when it is used, via TD(KD), in the radial distribution functions.

E (a) Defined near 20 as COSF (B4*YT) for quick computation of second energy term

(b) Defined near 38 as ET(KD), it acts as a dummy variable to ensure the pairing distances have the correct signs attached to them.

EN This is the contribution to the second energy term from an interacting K and KD. It is defined near 37 if K≠KD, and otherwise by 22.

ENC Input Data. Has a value $R/\sqrt{\pi}$, and must be subtracted from the K particle interaction energy ETE or EP, as in 31 or 32, to obtain the absolute energy including the third term given in equation (4.6.)

ENT Used in 22, but defined just after the data is read in, this variable saves time in that the calculation is then done only once.

ENY Defined by 31 or 32, this sums the interaction energies of the K particles over the assembly of NA2 particles and finally gives the cell energy of equation (4.6) in cell units.

EP Defined by 62 or 63, this sums the pairing energies between K and the KD's from the stored matrix, obtaining the permanent energy associated with K before it is moved.

ETE Defined after 35, this sums the newly calculated pairing energies between K and the KD particles. This temporary energy of interaction may become part of the total cell energy,ENY,if the attempt to move the particle is successful.

ET(KD) Defined by 35, this adds together the first and second energy terms after they have been summed over all the cells. It gives the interaction energy between K and KD. These temporary pairing energies, calculated for all the NA2 KD particles, hence the index, are stored in the pairing energy matrix if the K particle is moved.

F Defined near 20 as COSF (B4*ZT) for quick computation of the second energy term.

HD Defined near 21, this gives the value of the first energy term for one contribution only.

I A variable used in 159 to index the output.

JTAPE Input Data. Names the unit on which BARKER UNI TAPE No.8 is to be loaded.

K Designates the particle under consideration in the zero'th cell, and runs from 1 to 32. In formulae it is written k.

KD Designates the particle with which K interacts. Such particles exist in many cells including the zero cell. The accurate listing allows for 27 cells, whereas the fast listing allows for only KD in the zero cell, where KD is not allowed to equal K. In formulae it is written as k′.

KF Variable used in 160 to output the pairing energy matrix in convenient blocks.

KJ Variable used in 160 to output the pairing energy matrix in convenient blocks.

KTAPE Input Data. This names the unit on which NILSSON UNI TAPE No.3 is to be loaded.

LTAPE Input Data. This names the unit from which data is to be read.

M (a) Used in 191 an index to zero the pair distribution function.

(b) Used in 87 to bring the pairing distances to fixed point form (truncation) it then stores these distances in the pair distribution function.

ML(M) Defined by 82 as the like distribution function, this accumulates the number of like KD particles in the zero

cell at a distance M-1 tc M from the K particle. Then for each K particle considered, 15 like particles will have their pairing distance from K stored, note the distance between K and itself is neglected.

MU(M) Defined near 81 as the unlike distribution function, this acts as above, but for each K particle considered, 16 unlike particles will have their pairing distance from K stored.

MUC Used near 153 in the output of ML(M) and MU(M) to ensure the correct dimension of M. Also used in 191 if zeroing ML(M) and MU(M).

MTAPE Input Data. This names the unit on which output is to be written.

NA1 Input Data. This gives the number of positive particles present in the zero, or basic, cell.

NA2 Input Data. The total number of particles present in a unit cell. Variation of this parameter affects dimension, input, and output.

NA3 Input Data. Used near 104 with a value $(\frac{A4}{2} - 1)$, it is subtracted from NXD, NYD, NZD to form $\underline{\delta x}$, this displacement now being either positive or negative.

NDEL Defined near 30, this variable at 34 decides whether to move the K particle being considered, or whether a new K particle should be considered.

NQ (a) Defined by 50 and 51 to be positive for particles labelled from 1 to 16, and negative for particles labelled 17 to 32.

(b) Also used in 83 and 86 to ensure the pairing distances are read from the pairing energy-distance matrix.

NQD (a) Defined by 52 and 53 to be negative if unlike particles are interacting, and positive if like particles are interacting.

(b) Also used in 86 to ensure the pairing distances are read from the pairing energy-distance matrix.

NL Called the cell index, this sums over the cell index (l) of equation (2.7), and its limits are governed by the accuracy considerations of section 2.3. For the accurate listing, NL is given the value 0 by 54, then the contribution from the zero cell is calculated, and in 14 the next cell is considered. The computed go to statement just after 14 ensures each cell is given its correct coordinates, and then control again calculates that cells contribution to the energy. In the fast listing, where only the zero cell is considered, there is no need for a cell index and associated computation, and this saves time.

NR Defined near 101, this is used in the random number generator to obtain the integral part of R. See section 4.4 for discussion of random number generators.

NRN **Input** Data. This is the fixed point random number from the alternative random number generator, the merits of which are discussed in section 4.4. It is read in to obtain a full sequence of random numbers.

NUM Designated just before a random number is needed, this ensures that after the random number has been obtained via 101, that control comes back to the correct place in the program.

NUMBT A dummy variable used in 619 to transfer tape records, where use of variables on the record causes an error.

NUMB1 Input Data. This labels the iterations, and is essential if tape work is involved so that a particular iteration can be isolated. It adds one to itself each time control passes through 91, the commencement of the major loop.

NUMB2 Input Data. This controls (in 402), how many iterations are to be done by the computer in this run.

NUMB3 Input Data. This controls (in 403), how often output is to be printed.

NUMB4 Input Data. This checks (in 400), that the correct record has been read from the tape, if not it transfers control to the end of the program.

NUMB5 Input Data. This controls (near 403), when the first output is to be printed.

NUMB6 Input Data. This is used near 621 to isolate a particular record on a tape, usually for input purposes.

NUMB7 Input Data. Used in 409, this decided whether to zero or retain the pair distribution indices before commencing this run.

NUMB8 Input Data. Used near 625, this decides on the method of input, thus

<0, input data is read of cards and read in by a DO loop.

=0, tape records are transferred to the write tape, and the last of the read tape records is used as input.

>0, input obtained by reading one special record.

NUMB9 Input Data. Used in 619 to define the number of records to be transferred in the case where NUMB8=0.

NUM10 Input Data. Used in 591 this controls whether all output is written on tape, or only the last iteration is put on tape.

NUM12 Input Data. This in 564 controls whether a second tape should be loaded or not.

NXD
NYD
NZD Defined in 102, 103, 104, these give the three axial random coordinate displacements, and $\underline{\delta x^2}$ is given by $(NXD-NA3)^2 + (NYD-NA3)^2 + (NZD-NA3)^2$.

NX(K)
NY(K)
NZ(K) Input Data. These are the coordinates of the K^{th} particle in the zero cell, if the whole 32 are given, they completely define the configuration. They often are read in initially to start a run.

NXP
NYP
NZP — Defined by 28, these are used to retain the coordinates of the K^{th} particle in store while the program displaces K by $\underline{\delta x}$. If K prefers to remain in the permanent position, they are replaced as K's coordinates.

NXT
NYT
NZT — Defined by 18, 6 and 9, these give the axial distances between K and the KD in a particular cell, and are designed to include only the most important interactions, see section 4.1.

PDL — Input Data. This length is used to cut-off the pair distribution function. Due to the construction of the program it will lead to an error if it is greater than 500 mesh units, and this is its usual value.

PET — Defined by 29, this expression gives the probability of K going to the temporary energy configuration. The importance of this expression is discussed fully in section 2.2(b).

PE1(K,KD) — The pairing energy-distance matrix has dimension NA2×NA2. The elements above the diagonal store the interparticle distances between K and KD (in the zero cell only) while elements on and below the diagonal give the pairing energy between K and the KD (summed over the required number of cells). The elements are negative if unlike particles are interacting, and positive if like particles are interacting. The elements are formed in 39 and 40.

R Input Data. This is the random number last used in the first random number generator, and is input to ensure a full sequence of random numbers is generated by 101.

RN Used to convert NR into a floating point number, so it can be subtracted from R.

RTEM Used in the alternatively listed fixed point random number generator to convert NRN into floating point number, prior to division by a floating point number.

SAO The square of AO, this is used in 25 in the alternative listing to save computing time in taking square roots.

SUMEN Defined near 15, this gives the second energy term contribution for a particular K KD interaction in the zero cell.

SUMHD Defined near 16, this sums the first energy term contributions for the various cells for a particular K and KD.

TD(KD) Defined by 71, this stores the interparticle distances between the moved K particle, and the other particles in the zero cell. If the temporary configuration is preferred, these interparticle distances are placed in the pairing energy matrix by 38.

TNA5 Input Data. Used near 87, this enables the pair distribution function to be divided into coarse or

fine divisions. Its usual value is 1 to obtain the finest calibration of the pair distribution function possible.

TSMEN Used just after 35, this adds together all the SUMEN to obtain the second energy term for a K particle and all the KD particles.

TSMHD Used after 35, this adds together SUMHD to obtain the first energy term for a K particle and all the KD particles.

XT
YT
ZT Used at 12 to convert NXT, NYT and NZT into floating point form preparatory to calculation.

4.4 Subprogramme

During the calculation, only one subroutine is used, which is CALL DNSHI(JTAPE). This subroutine is standard for almost all computers using tape units, and its function is to ensure that the tape is written on in high density form rather than low density, which is sometimes used for transferring rather than storing data. The subroutine does not need to be provided, as it is written into the machine.

Several functions are used in the calculation, specifically COSF, SQRTF, EXPF, ERRF and RANDF. The Cosine function, COSF, square root function, SQRTF, and exponential function EXPF, are all standard functions and are built into the machine. However, the Error function ERRF and Random number function RANDF needed to be built into the program, although they could be obtained in binary form from the 7090 share library, and will be discussed in some detail.

ERRF. This evaluates $E(X) = \frac{2}{\sqrt{\pi}} \int_0^X e^{-t^2} dt$, often called the error integral, probability integral or integrale de Gauss, by using Hastings approximation

$ERRF(X) \simeq 1-(\sum_{i=0}^{b} a_i t^i)^{-16}$; where

$a_0 = 1.0000000$, $a_1 = .070523078$, $a_2 = .042282012$

$a_3 = .0092705272$, $a_4 = .0001520143$, $a_5 = .0002765672$,

a_6 = .0000430638 and X must be a floating point number such that $0 \leq X < \infty$, and then the error will be less than 3×10^{-7}. The time given to calculate one value of Y = ERRF(X) is 3.6 millisec on the 704 computer, which means this is the really time-consuming operation in the program, and as mentioned in section 2.3 there are several ways of avoiding it, but only with loss of accuracy. Several forms of calculation E(X) were tried, and the above form as given in IBM share program A A ERF2(C3) was found to be the fastest.

RANDF. As the theory has stressed, random numbers play an important part in a Monte Carlo calculation, and hence must be carefully chosen with the least possible bias. Perhaps one of the best methods of generating random numbers is the power residue method, which is particularly convenient on modern computers. This has the form

$$R_{n+1} = (K R_n + c) \bmod p,$$

and has been applied to the 7090 computer by Rotenburg[1], where he chooses the K,c and p to reduce correlation between successive random numbers. The critereon for small correlation is given by Coveyou[2]. To suit the machine P is chosen as 2^{35}, and a choice of $K = 2^7+1$ and $c = (.5 \pm \frac{\sqrt{3}}{6})\, 2^{35}$ minimises the correlation between successive

random numbers. Most computers have a random number generator routine, either built in, or readily accessible from their library; however in this case the random number from the library function was obtained by calling RANDF(ARG) and the sequence started at a given point. When the program was removed from the computer, the random number function reset itself to start from the same given point when next called, and as the calculation was to be done in sections this meant that each section was started from the same point in the sequence, and this was unsatisfactory. To obtain a continuous sequence the following program was devised.

From

$$R_{n+1} = [(2^7+1)R_n + (.5+\frac{\sqrt{3}}{6})2^{35}] \bmod 2^{35} \qquad (4.1)$$

we have $\frac{R_{n+1}}{2^{35}} = \left[129\,\frac{R_n}{2^{35}} + .7886751346\right] \bmod 1$, and in program form this is

```
101     R = 129.*R + .7886751346
       NR = R
       RN= NR
        R = R-RN
```

Whenever a random number was needed the program defined NUM, went to 101 and calculated the random number R, then returned with it to the relevant part of the program nominated by NUM.

The random numbers so generated are distributed rectangularly on (0,1), and appear to obey the simpler randomness tests well, but they do not follow the sequence (4.1) as the 7090 computer only carries 10 figures, and the rest in 101 are truncated. It is difficult to tell to what extent the sequence is affected, whether the period is excessively reduced or the correlation increased, and the method was deemed sufficient for the shorter runs.

For longer runs an alternative generator was devised, which had a definite period and correlation. It works in fixed point mode and is based on the characteristic of the 7090 computer to take any fixed point number as modulo 2^{17}, (131,072), and this is the period. The power residue form is used, with constants and form chosen similarly to the previous generator, so

```
101 NRN = (129*NRN) + 103 373
    RTEM = NRN
    R = RTEM/131,071
```

NRN gives the fixed point random number, which becomes distributed on (0,1) by the division of 131,071, and so NRN is read in at the commencement of every run, to ensure a continuous sequence. This routine is fast and convenient, has a period sufficient to deal with over 1000 large iterations, and has the additional advantage that similar sequences can easily be generated on smaller computers.

REFERENCES TO CHAPTER IV

[1] Rotenburg,A. Jour. of the Assoc. for Computing Machinery Vol.7 p.75. (1960).

[2] Coveyou,R.R. Jour. of the Assoc. for Computing Machinery Vol.7 p.73. (1960).

APPENDIX I

UNITS

During computation the problem is most conveniently considered in terms of the '<u>cell unit</u>', since this gives the parameter R a value approximately 2. The cell unit is further divided into '<u>mesh units</u>' which enables much of the arithmetic to be done in fixed point mode. The cell unit = 1000 mesh units, and its value is determined by the density to be considered. Thus sixteen electrons in a cube of 1×1×1 cubic cell units must be equivalent to the electron number density N.

$$\frac{16}{1\text{c.cell unit}} = \frac{N}{\text{cc}} \text{ electrons}$$

$$1 \text{ cell unit} = \sqrt[3]{\frac{16}{N}} \text{ cm}$$

$$= 2.52\times10^{-6} \text{cm for an electron density of } 10^{18} \text{e/cc.}$$

The energy calculated, E_c, is computed in terms of (cell units length)$^{-1}$. Hence energy in

$$\text{ergs} = \frac{(\text{statcoulomb})^2}{\text{cm}} = \frac{E_c \times e^2}{\text{conversion factor from cell units to cm.}}$$

$$\therefore \text{ Energy in ergs} = \frac{23.06746\times10^{-20}}{\sqrt[3]{16/N}} \text{ E in cell units.}$$

The Boltzmann constant can now be converted into cell units

$$k = 1.38044\times10^{-16} \text{ erg/°K}$$

$$= \frac{1.38044\times10^{-16} \times \sqrt[3]{16/N}}{23.06746\times10^{-20}} \text{ cell units/°K}$$

and $\beta = \frac{1}{kT} = 16.7102 \times \frac{10^{-4}}{2.52} \times \frac{\sqrt[3]{N}}{T}$.