INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE: CENTRATOM TRIESTE

H4.SMR/854-22

College on Computational Physics

15 May - 9 June 1995

Molecular Dynamics Simulations

F. Ercolessi

S.I.S.S.A.
Trieste, Italy

MAIN BUILDING STRADA COSTIERA, 11 TEL. 2240111 TELEFAX 224163 TELEX 460392 ADRIATICO GUEST HOUSE VIA GRIGNANO, 9 TEL. 224241 TELEFAX 224531 TELEX 460449
MICROPROCESSOR LAB. VIA BEIRUT, 31 TEL. 224471 TELEFAX 224600 TELEX 460392 GALILEO GUEST HOUSE VIA BEIRUT, 7 TEL. 22401 TELEFAX 2240310 TELEX 460392

Furio Ercolessi

MOLECULAR DYNAMICS SIMULATIONS

College on Computational Physics
ICTP - Trieste
May - June 1995

See also URL http://www.sissa.it/furio/md/
on the World Wide Web

Molecular dynamics simulations

- Introduction: method, applications
- How to program it
- How to extract useful data
- How to go fast
- WAIT A MOMENT !
  What are we doing ?
  [Interatomic potentials]

Good books on molecular dynamics

- M.P. Allen and D.J. Tildesley
  Computer simulation of liquids
  Oxford Univ. Press, 1987
  [available in paperback]

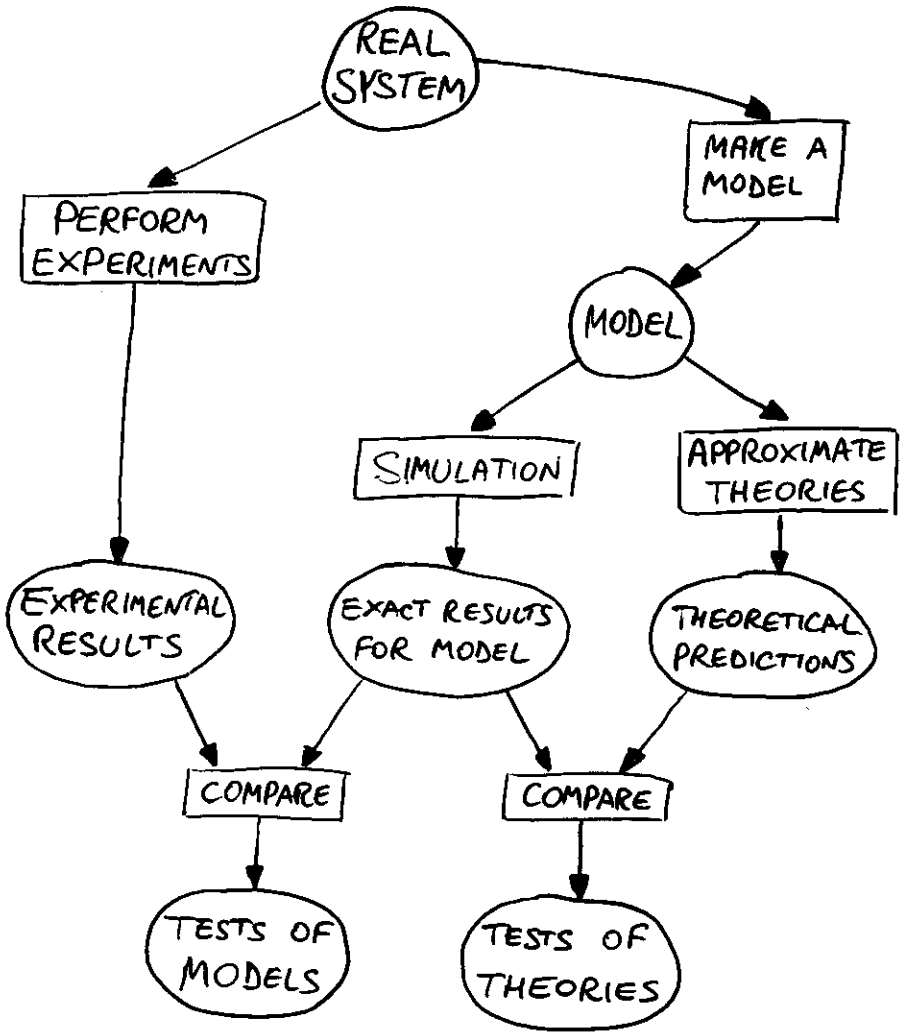
- J.M. Haile
  Molecular dynamics simulation
  Wiley, 1992

Programs in Fortran 90 accompanying this course

[www: http://www.sissa.it/Furio/md/f90/]

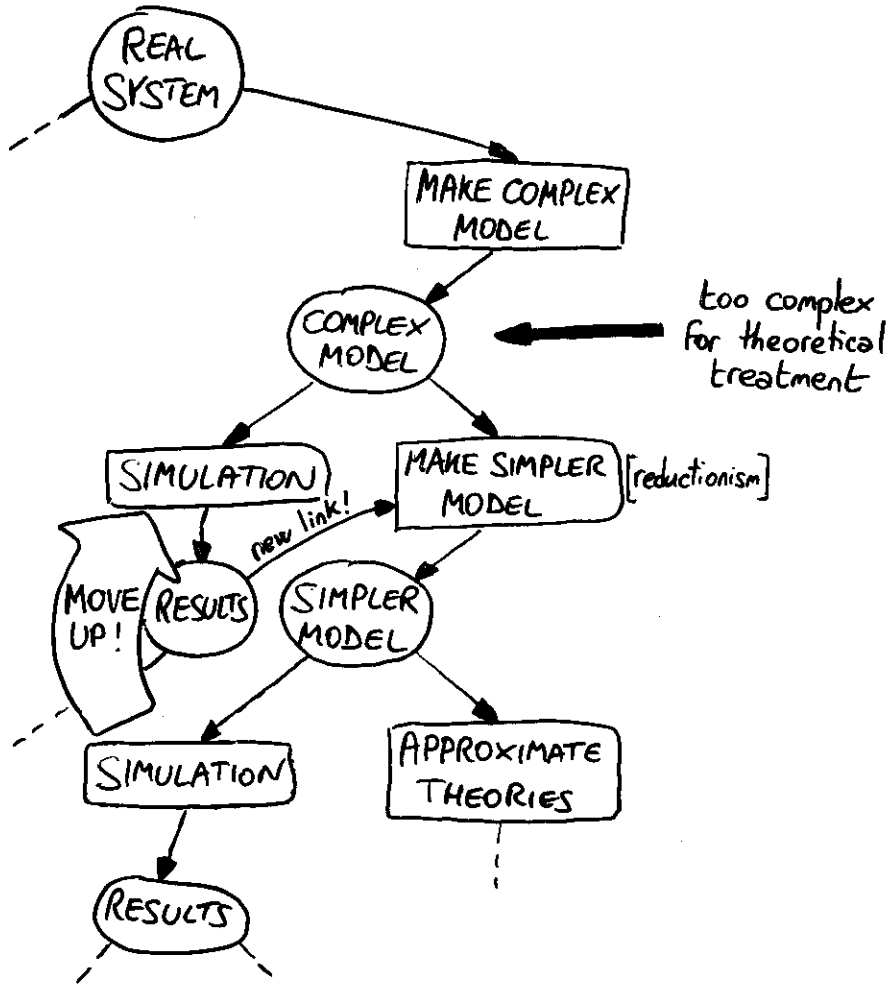
crystal.f90	to generate initial configurations with atoms arranged in an fcc crystal structure
md1.f90	molecular dynamics, Lennard-Jones potential computed "analytically"; slow double loop over all pairs to compute the force
md2.f90	as md1, but the LJ potential is tabulated numerically, and the calculation proceeds by table look-ups
md3.f90	as md2, but with Verlet neighbor lists with automatic updating
md3glue.f90	as md3, but for a many-body "glue" potential. The file aluminum.f, to be compiled together with md3glue, contains the functions for the ab-initio-derived Al potential

The "traditional" role of simulation

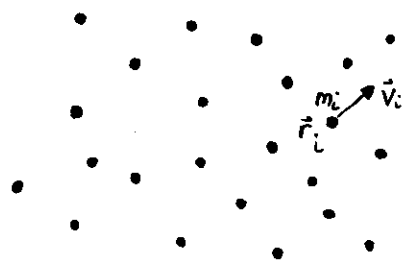


New roles : allow for more complex models

[an alternative to reductionism]



Molecular Dynamics



- Set of N atoms, treated as "point" particles
- Use classical mechanics
- Assign masses m_i, positions $\vec{r}_i(t=0)$, velocities $\vec{v}_i(t=0)$
- Time evolution by integrating Newton's equations:

$$\vec{F}_i = m_i \frac{d^2\vec{r}_i}{dt^2}$$

Getting the forces $\vec{F}_i$

- Define potential energy function

$$V(\vec{r}_1, \vec{r}_2, \ldots, \vec{r}_N)$$

This is the model

- Force acting on atom i in configuration $\{\vec{r}_1, \vec{r}_2, \ldots \vec{r}_N\}$:

$$\vec{F}_i = -\vec{\nabla}_{\vec{r}_i} V(\vec{r}_1, \ldots, \vec{r}_N)$$

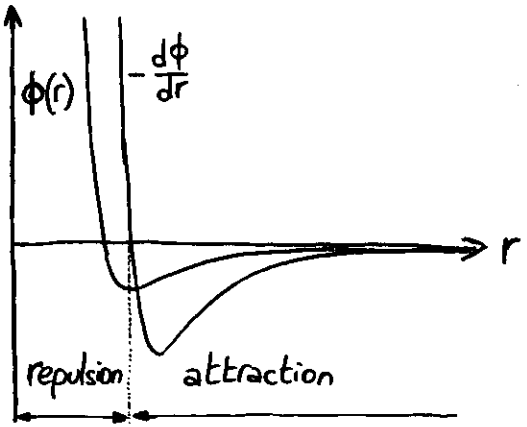
- Simplest possibility (but not the only one!):

$$V(\vec{r}_1, \ldots, \vec{r}_N) = \sum_i \sum_{j>i} \phi(|\vec{r}_i - \vec{r}_j|)$$

implying

$$\vec{F}_i = -\sum_{j \neq i} \left.\frac{d\phi}{dr}\right|_{|\vec{r}_i - \vec{r}_j|} \frac{\vec{r}_i - \vec{r}_j}{|\vec{r}_i - \vec{r}_j|}$$

A typical pair potential



Example: Lennard-Jones potential

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

repulsion attraction

- Pair potentials only good for closed-shell systems (rare gases)
- Analytical form not strictly necessary
- Many-body potentials available for metals and semiconductors

Time integration:

by Finite difference methods

$\vec{r}_i(t)$ $\quad\Rightarrow\quad$ $\vec{r}_i(t+\Delta t)$

$\vec{v}_i(t)$ $\quad\Rightarrow\quad$ $\vec{v}_i(t+\Delta t)$

$\vec{a}_i(t) = -\frac{1}{m_i}\vec{\nabla}_i V(\vec{r}_1(t)\ldots\vec{r}_N(t))$ $\quad\Rightarrow\quad$ $\vec{a}_i(t+\Delta t) = \ldots$

Verlet' algorithm:

$$\vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 + \frac{1}{6}\vec{b}(t)\Delta t^3 + O(\Delta t^4)$$

$$\vec{r}(t-\Delta t) = \vec{r}(t) - \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 - \frac{1}{6}\vec{b}(t)\Delta t^3 + O(\Delta t^4)$$

$$\vec{r}(t+\Delta t) = 2\vec{r}(t) - \vec{r}(t-\Delta t) + \vec{a}(t)\Delta t^2 + O(\Delta t^4)$$

Computing properties From trajectories

Molecular dynamics

$$\langle A \rangle = \lim_{t\to\infty} \frac{1}{t}\int_{t_0}^{t_0+t} A(\xi)\,d\xi$$

$$= \lim_{N_T\to\infty} \frac{1}{N_T}\sum_{1}^{N_T} A(k\Delta t)$$

t (or N_T) are Finite in practice.

Also, the system size is Finite.

Statistical mechanics

$$\langle A \rangle = \frac{\int\cdots\int e^{-\beta H(\Gamma)} A(\Gamma)\,d\Gamma}{\int\cdots\int e^{-\beta H(\Gamma)}\,d\Gamma}$$

$\beta = 1/k_B T \quad \Gamma = \{\vec{r}_1 \cdots \vec{r}_N, \vec{p}_1 \cdots \vec{p}_N\}$

Should be the same according to the ergodic hypothesis

Isolated system $\Rightarrow$ Conservation laws

- $E = K + V$

  $K(t) = \frac{1}{2}\sum_i m_i v_i^2(t)$

  $V(t) = V(\vec{r}_1(t), \ldots, \vec{r}_N(t))$

- $\vec{P} = \sum_i m_i \vec{v}_i(t)$

- $\vec{L} = \sum_i m_i [\vec{r}_i(t) \times \vec{v}_i(t)]$ (broken by PBCs)

Molecular dynamics methods

CLASSICAL	FIRST PRINCIPLES (CAR-PARRINELLO)
$V = V(\vec{r}_1, \dots \vec{r}_N)$	$V = \min_\psi E_{LDA}[\psi, R]$ (DFT-LDA)
Potential ?	Accurate
$N \sim 10^4 - 10^7$	$N \sim 10^2 - 10^3$
$t \sim$ ns	$t \sim$ ps
$O(N)$ easy	$O(N)$ in the works

Many problems of practical interest have to be done classically

Can we use classical forces ?

$$m\bar{v} = \frac{2\pi\hbar}{\Lambda} \qquad \frac{1}{2}m\bar{v}^2 \sim \frac{k_B T}{2}$$

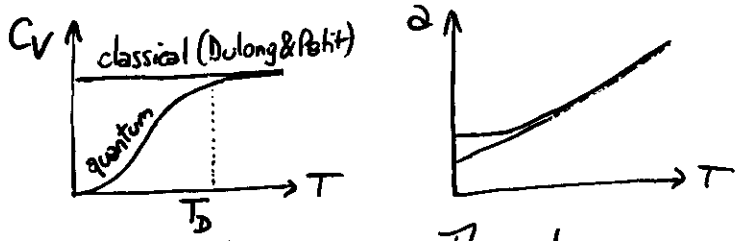
$$\Lambda = \sqrt{\frac{2\pi\hbar^2}{m k_B T}}$$ De Broglie wavelength

$\frac{\Lambda}{a} \ll 1$: classical Ok

$\frac{\Lambda}{a} \sim 1$: quantum effects

Liquids at the triple point : { Li, Ar and heavier Ok
H_2, He, Ne quantum fluids

Quantum effects always appear in solids for T low enough



EXAMPLES : Specific heat Thermal expansion

FIRST MD PAPER J. Chem. Phys. 27, 1208 (1957)

Phase Transition for a Hard Sphere System

B. J. ALDER AND T. E. WAINWRIGHT
University of California Radiation Laboratory, Livermore, California
(Received August 12, 1957)

A CALCULATION of molecular dynamic motion has been designed principally to study the relaxations accompanying various nonequilibrium phenomena. The method consists of solving exactly (to the number of significant figures carried) the simultaneous classical equations of motion of several hundred particles by means of fast electronic computors. Some of the details as they relate to hard spheres and to particles having square well potentials of attraction have been described.[1,2] The method has been used also to calculate equilibrium properties, particularly the equation of state of hard spheres where differences with previous Monte Carlo[3] results appeared.

The calculation treats a system of particles in a rectangular box with periodic boundary conditions.[4] Initially, the particles are in an ordered lattice with velocities of equal magnitude but with random orientations. After a very short initial run[1,2] the system reached the Maxwell-Boltzmann velocity distribution so that the pressure could thereafter be evaluated directly by means of the virial theorem, that is by the rate of change of the momentum of the colliding particles.[1,2] The pressure has also been evaluated from the radial distribution function.[5] Agreement between the two methods is within the accuracy of the calculation.

A 32-particle system in a cube and initially in a face-centered cubic lattice proceeded at about 300 collisions an hour on the UNIVAC. For comparison a 96-particle system in a rectangular box and initially in a hexagonal arrangement has been calculated, however only at high densities so far. No differences in the pressures can be detected. It became apparent that some long runs were necessary at intermediate densities, accordingly the IBM-704 was utilized where, for 32 particles, an hour is required for 7000 collisions. Larger systems of 108, 256, and 500 particles can also conveniently be handled; in an hour 2000, 1000, and 500 collisions, respectively, can be calculated. The results for 256 and 500 particles are not now presented due to inadequate statistics.

The equation of state shown in Fig. 1 of the accompanying paper[6] for 32 and 108 particles is for the intermediate region of density, where disagreement was found with the previous Monte Carlo results. The volume, v, is given relative to the volume of close packing, v_0. Plotted also are the more extended Monte Carlo results; the agreement between these three systems is within the present accuracy of the pressure determination. This agreement provides an interesting confirmation of the postulates of statistical mechanics for this system.

Figure 1 of the accompanying paper shows two separate and overlapping branches. In the overlapping region the system can, at a given density, exist in two states with considerably different pressures. As the calculation proceeds the pressure is seen to jump suddenly from one level to the other. A study of the positions of the particles reveals that as long as the system stays on the lower branch of the curve the particles are all confined to the narrow region in space determined by their neighbors, while on the upper branch of the curve the particles have acquired enough freedom to exchange with the surrounding particles. Since the spheres are originally in ordered positions, the system starts out on the lower branch; the first jump to the upper branch can require very many collisions. The trend, as expected, is that at higher densities more collisions are necessary for the first transition, however, there are large deviations. At $v/v_0=1.60$, 5000 collisions were required; at 1.55, 25 000; while at 1.54 only 400; at 1.535, 7000; at 1.53, 75 000; and at 1.525, 95 000. Runs in excess of 200 000 collisions at v/v_0 of 1.55 and 1.53 have not shown any return to the lower branch, while at 1.525 the system has returned several times, however only for relatively few collisions. The lowest density at which the system did not jump to the upper curve is at 1.50, however the run extends only to 50 000 collisions and at that density it might take very many collisions before the appropriate fluctuation occurs for a molecule to escape from its neighborhood. For comparison, the first jump for 108 particles occurred for $v/v_0=1.55$ and 1.60 at about 2000 collisions. This is fewer collisions per particle than for the smaller system and is indicative of larger possible density fluctuations in larger systems. Apparently, the reason these jumps occur rather than the system's taking on some mean pressure is that it is not possible to have the two states simultaneously in equilibrium in such a small sample. The fact that the 108 and 32 particles systems give identical branches indicates that the effect of the periodic boundary is not serious even for 32 particles. It can be shown that the first few virial coefficients for a finite system of N particles with periodic boundary conditions have corrections of order $1/N$.

Clearly, it is presently impossible to connect these two branches, though it may be possible to do this with a faster machine by averaging the pressure over a very long run which includes many jumps instead of averaging the branches separately. In the transition region the effect of the number of particles might appear in the average time the system spends in each branch, such that for an infinite system a horizontal line might appear, characteristic of the first-order transition which is strongly indicated by the present results. Such a transition was suggested earlier by the superposition theory at a $v/v_0=1.48$.

We gratefully acknowledge Mrs. Shirley Campbell's and Mary Shephard's help with the coding problems.

[1] Proceedings of the International Union of Pure and Applied Physics on "Statistical mechanical theory of transport properties," Brussels (1956).
[2] Symposium on the "Many body problem" in New York (1957).
[3] M. N. Rosenbluth and A. W. Rosenbluth, J. Chem. Phys. 22, 881 (1954).
[4] Alder, Frankel, and Levinson, J. Chem. Phys. 23, 417 (1955).
[5] Kirkwood, Maun, and Alder, J. Chem. Phys. 18, 1040 (1950).
[6] W. W. Wood and J. D. Jacobson, J. Chem. Phys. 27, 1207 (1957).

PHYSICAL REVIEW VOLUME 120, NUMBER 4 NOVEMBER 15, 1960

Phys. Rev. 120, 1229 (1960)

Dynamics of Radiation Damage*

J. B. GIBSON, A. N. GOLAND,† M. MILGRAM, AND G. H. VINEYARD
Brookhaven National Laboratory, Upton, New York

(Received July 14, 1960)

Radiation damage events at low and moderate energies (up to 400 ev) are studied by machine calculations in a model representing copper. Orbits of knock-on atoms are found and the resulting damaged configurations are observed to consist of interstitials and vacancies. Thresholds for producing permanently displaced atoms (i.e., interstitials) are about 25 ev in the ⟨100⟩ direction, 25 to 30 ev in the ⟨110⟩ direction, and around 85 ev in the ⟨111⟩ direction. Collision chains in the ⟨100⟩ and ⟨110⟩ directions are prominent; at low energies the chains focus, at higher energies they defocus. Above threshold, the chains transport matter, as well as energy, and produce an interstitial at a distance. The range of ⟨110⟩ chains has been studied in detail. Localized vibrational modes associated with interstitials, agitations qualitatively like thermal spikes, ring annealing processes, and a higher energy process somewhat like a displacement spike have been observed. Replacements have been found to be very numerous.

The configurations of various static defects have also been studied in this model. The interstitial is found to reside in a "split" configuration, sharing a lattice site with another atom. The crowdion is found not to be stable, and Frenkel pairs are stable only beyond minimum separations, which are found to be very much dependent on orientation.

I. INTRODUCTION

THE initial event in the damaging of a crystal lattice by high-energy radiation is the sudden transfer of a rather large amount of kinetic energy (10 to perhaps 10^5 ev) to a single atom. The energized atom then ploughs through the lattice knocking other atoms from their sites and leaving a damaged region behind. From a theoretical standpoint this damaging event is a complex many-body problem, and it has been treated in the past only by making drastic approximations.[1] Generally it has been considered as a cascade of independent, two-body collisions between knock-on atoms and stationary atoms. The knock-on atoms have been assumed to move freely between collisions. The stationary atoms have been assumed to behave as though randomly located, and their binding in the lattice has been taken into account by the very much simplified assumption that they will be displaced and enter the group of freely moving knock-ons if and only if endowed with energy above a certain threshold, generally in the neighborhood of 25 ev. On this cascade model the damage is predicted to be a set of interstitial atoms and an equal number of vacant lattice sites, distributed randomly over a small region. Other models have been proposed in which many-body effects are given prominence. Thermal spike and displacement spike models are of this character. In the former, the region around the site of a knock-on is assumed to behave as if suddenly heated, and its subsequent cooling is treated by the classical laws of heat conduction in a homogeneous medium. In the displacement spike models, qualitative arguments about the character of damage are advanced on the assumption that a kind of miniature "explosion" occurs around the site of the knock-on. These models are difficult to harmonize with one another, and each has obvious shortcomings. Patchwork attempts at improving the models in individual details have not yet been very impressive.

In the last few years a number of sophisticated radiation damage experiments have been made. In the most notable of these highly purified metals have been

* Work supported by the U. S. Atomic Energy Commission.

† Guest Scientist from Ordnance Materials Research Office, Watertown, Massachusetts.

[1] For reviews see F. Seitz and J. S. Koehler, in *Solid-State Physics*, edited by F. Seitz and D. Turnbull (Academic Press, Inc., New York, 1956), Vol. 2, p. 305; also G. J. Dienes and G. H. Vineyard, *Radiation Effects in Solids* (Interscience Publishers, Inc., New York, 1957).

Figure 20 shows the z-component of displacement for more shots along ⟨111⟩, first 25 volts with interatomic potential number 1, then 25 and 40 volts with interatomic potential number 2. In the first of these the struck atom does not even penetrate the triangle of near neighbors, and is clearly returning directly to its site, even though the calculation was run only a very short time. The effect of increasing the atomic size is evident—the threshold for reaching the center of the cube has become much higher. The shot at 25 ev with potential 2 is very much like that at 20 ev with potential 3 (Fig. 19), and is also below threshold. That at 40 ev with potential 2 is like that at 30 ev with potential 3.

Two higher energy shots are shown in Fig. 21. Both involve interatomic potential 2. In both cases the struck atom was 2,2,2 and its initial motion was along [111]. Δz is plotted against time for atoms 2,2,2, 4,4,4, and 6,6,6. In one shot (dotted lines in Fig. 21) the initial kinetic energy was 60 ev, in the second (solid lines) it was 100 ev. It is seen that 60 ev is only enough to project the interstitial into the nearest cube-center position, whence it must decay by the previously mentioned mechanism, resulting in two replacements and no displacements. With 100 ev, an interstitial is created temporarily at the center of the second cube (5,5,5). In times beyond the end of the run this should produce an interstitial in the split configuration, whose exact location will be determined by any slight departures from symmetry; each of the six possible locations is known from static calculations to be stable. The threshold for production of a permanently displaced atom by a shot in the direction ⟨111⟩ (with interatomic potential 2) is thus seen to be between 60 and 100 ev, a value notably higher than for the directions ⟨100⟩ and ⟨110⟩.

The next three figures show results of some shots above threshold in a direction well away from symmetry axes, namely 10° away from [011] in the plane $x=2$. In the first of these the atomic set B was used, in the last two the large set C was used (see Table I). Orbital plots of the plane $x=2$ are shown. In all cases the

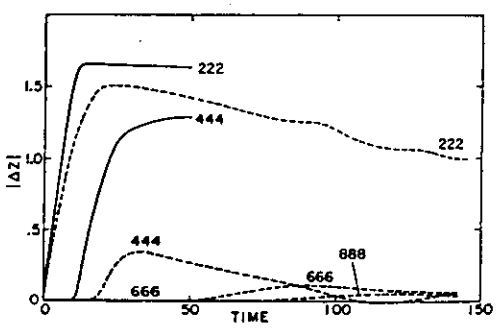


FIG. 21. Two ⟨111⟩ shots at higher energies with potential 2. Δz vs time is shown for several atoms. Dotted lines, knock-on energy 60 ev (Run No. 69); solid line, knock-on energy 100 ev (Run No. 70).

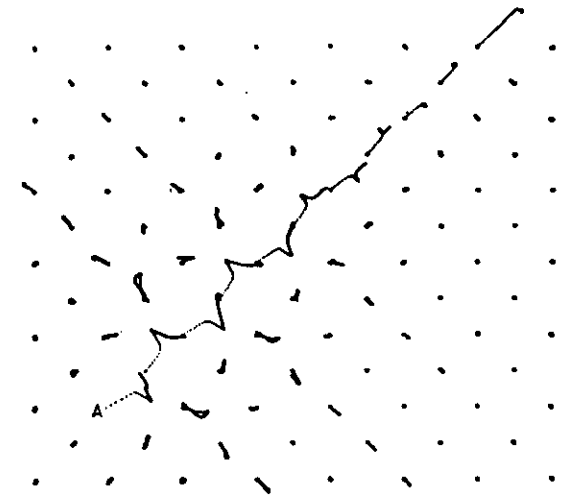


FIG. 22. Shot in (100) plane at 50 ev, 10° away from [011]. Orbits in (100) plane, to time 92, are shown. Knock-on started at A (Run No. 71).

interatomic potential was number 2. In Fig. 22 the initial kinetic energy was 50 ev. A collision chain is seen, which leaves a vacancy at its beginning (marked A) and is about to produce an interstitial after perhaps 10 replacements. The transition from a defocused to a focused condition is evident in this chain. Figure 23 shows a 100-ev shot (No. 96) which, because of the large set in which it was run is completely contained. Time runs up to about 200 units in this figure. The knock-on atom was at V_1. Two vacancies are created, at V_1 and V_2, and

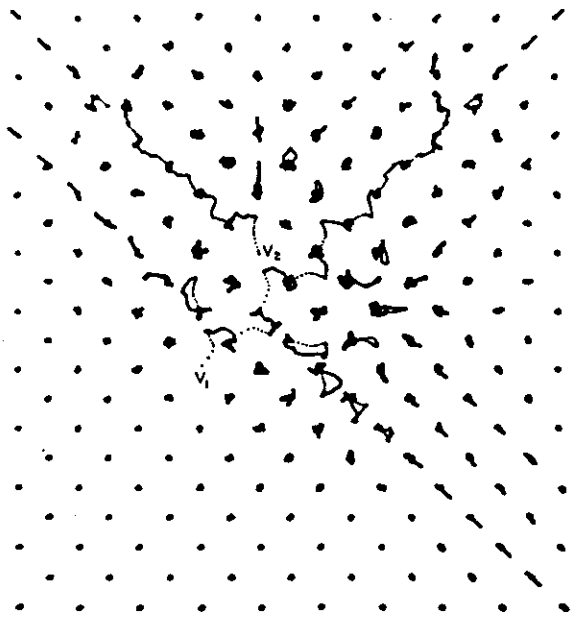


FIG. 23. Shot in (100) plane at 100 ev, 10° away from [011]. Orbits in plane to time 200 (approximately) are shown. Knock-on started at V_1 (Run No. 96).

Notes on p. 59

PHYSICAL REVIEW VOLUME 136, NUMBER 2A 19 OCTOBER 1964

(1964)

Correlations in the Motion of Atoms in Liquid Argon*

A. RAHMAN
Argonne National Laboratory, Argonne, Illinois
(Received 6 May 1964)

A system of 864 particles interacting with a Lennard-Jones potential and obeying classical equations of motion has been studied on a digital computer (CDC 3600) to simulate molecular dynamics in liquid argon at 94.4°K and a density of 1.374 g cm^{-3}. The pair-correlation function and the constant of self-diffusion are found to agree well with experiment; the latter is 15% lower than the experimental value. The spectrum of the velocity autocorrelation function shows a broad maximum in the frequency region $\omega=0.25(k_BT/\hbar)$. The shape of the Van Hove function $G_s(r,t)$ attains a maximum departure from a Gaussian at about $t=3.0\times10^{-12}$ sec and becomes a Gaussian again at about 10^{-11} sec. The Van Hove function $G_d(r,t)$ has been compared with the convolution approximation of Vineyard, showing that this approximation gives a too rapid decay of $G_d(r,t)$ with time. A delayed-convolution approximation has been suggested which gives a better fit with $G_d(r,t)$; this delayed convolution makes $G_d(r,t)$ decay as t^4 at short times and as t at long times.

I. INTRODUCTION

IN recent years considerable use has been made of large digital computers to study various aspects of molecular dynamics in solids, liquids, and gases.[1] The following is a description of a computer experiment on liquid argon (using the CDC 3600) to study the space and time dependence of two-body correlations which determine the manner in which slow neutrons are inelastically scattered from the liquid. If neutron scattering data of unlimited accuracy and completeness was available, then the kind of work presented here would serve the useful though unexciting purpose of confirming the results already obtained with neutrons. At present, however, the situation is that theorists are trying to build models for these two-body dynamical correlations to account for the observed neutron spectra; the current interest in the work presented here is thus to throw some light on the validity of these models, and to suggest the manner in which some improvements can be made.

The calculations presented here are based on the assumption that classical dynamics with a two-body central-force interaction can give a reasonable description of the motion of atoms in liquid argon. For practical reasons, further assumptions have to be made, namely, the interaction potential has to be truncated beyond a certain range, the number of particles in the assembly has to be kept rather small, and suitable boundary conditions have to be imposed on the assembly. Finally, the equations of motion have to be solved as a set of difference equations, thus involving a certain increment of time to go from one set of positions and velocities to the next. The details will be set forth in the next section. At the end of the paper a brief mention will be made of checks on the validity of these assumptions. The results presented in this paper are confined mainly to one pair of values of the temperature and the density of the system, namely, 94.4°K and 1.374 g cm^{-3}. A less exhaustive study, at 130°K and 1.16 g cm^{-3}, is mentioned briefly at the end.

II. METHOD OF COMPUTATION

The calculations reported here were based on the following ingredients.

Particles with mass $39.95\times1.6747\times10^{-24}$ g (the mass of an argon atom) were assumed to interact in pairs according to the potential $V(r)=4\epsilon\{(\sigma/r)^{12}-(\sigma/r)^{6}\}$, $\epsilon/k_B=120$°K, $\sigma=3.4$ Å, r being the distance between the particles. This interaction was assumed to extend up to a range $R=2.25\sigma$, so that a particle interacts with all particles situated within a sphere of that radius; $V(2^{1/6}\sigma)=-\epsilon$ is the minimum of $V(r)$ and at $r=R$, $V\sim-0.03\epsilon$.

864 such particles were placed in arbitrary positions in a cubical box of side $L=10.229\sigma$, thus providing a density of 1.374 g cm^{-3}. Periodic boundary conditions were imposed, so that at any given moment a particle with coordinates x, y, z inside the real box implied the presence of 26 periodic images with coordinates obtained by adding or subtracting L from each Cartesian coordinate. The density was conserved because when a particle moves out across one face of the cube another moves in across the opposite face.

The particles were then allowed to move, and their motions were calculated using a set of difference equations with a time increment of 10^{-14} sec. The details have been given in an Appendix. The positions and velocities obtained at successive moments were recorded on magnetic tape for later analysis. The only quantity monitored during the progress of the calculation was the mean-square velocity of the particles expressed in temperature units,

$$T=\frac{M}{3Nk_B}\sum_{i=1}^{N}\mathbf{v}_i^2,$$

where $N=864$. In the initial stages of the calculation, if T was not in the region of temperature (90°K) at which the system was to be studied, all velocities were

* Based on work performed under the auspices of the U. S. Atomic Energy Commission.

[1] J. R. Beeler, Jr., in *Physics of Many-Particle Systems*, edited by E. Meeron (Gordon and Beach Publishers, Inc., New York, 1964).

Today:

Materials science applications

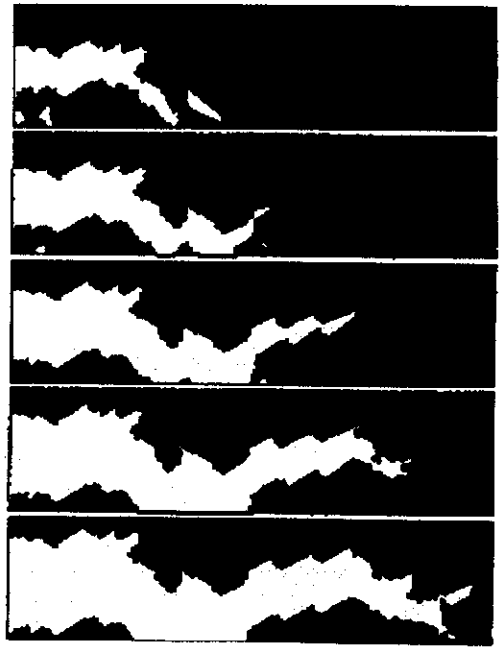
Fracture of 2D notched solids under tension
F.F. Abraham et al. Phys. Rev. Lett. 73, 272 (1994)
IBM Almaden www: http://www-i.almaden.ibm.com/vis/mol_dyn.html

~ 10^6 atoms
Parallel computer

F.F. Abraham et al.
IBM Almaden

3a

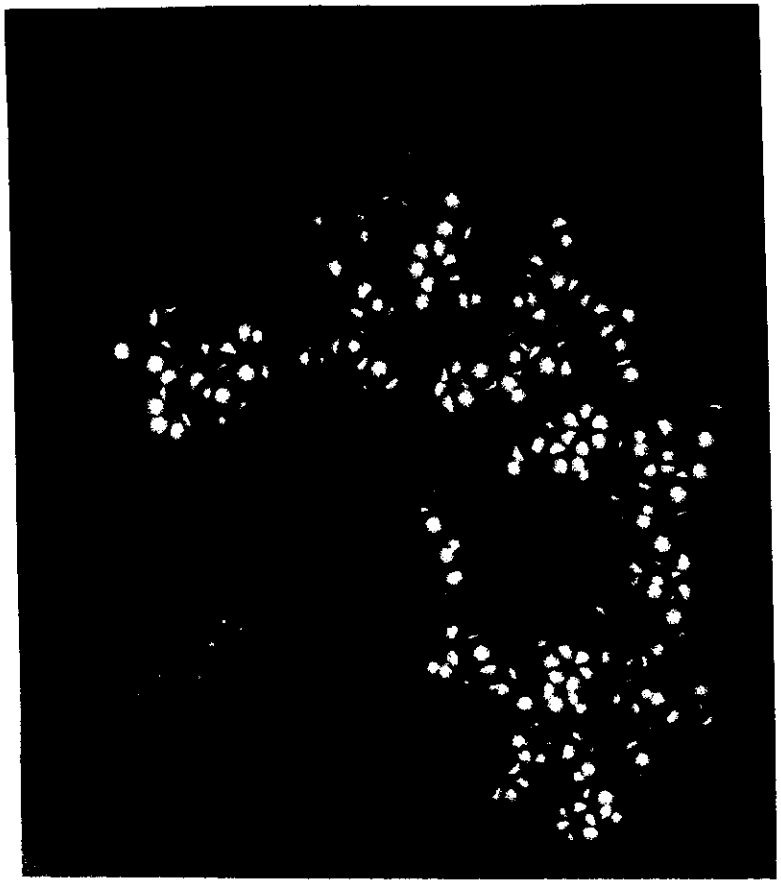
Dynamic instability of crack tip



3b

Biomolecules

Proteins, nucleic acids (DNA, RNA), ... etc
Drug design applications



Photosynthetic Reaction Center of Rhodopseudomonas Viridis, H. Treutlein and K. Schulten, U. of Münich/NCSA

"Biological solar battery": sun light ⇒ polarization

Measuring simple quantities

GENERAL RULE: $\langle A \rangle = \frac{1}{N_T} \sum_{1}^{N_T} A(k\Delta t)$

where $A(t) = A\left(\vec{r}_1(t), \ldots, \vec{r}_N(t), \vec{v}_1(t), \ldots, \vec{v}_N(t)\right)$

POTENTIAL ENERGY: $\langle V(\vec{r}_1 \ldots \vec{r}_N) \rangle = \langle V \rangle$

KINETIC ENERGY: $\left\langle \frac{1}{2} \sum_i m_i v_i^2 \right\rangle = \langle K \rangle$

TOTAL ENERGY: $\langle E \rangle = \langle K \rangle + \langle V \rangle = E = \text{constant}$

TEMPERATURE: Assume equipartition ⇒

$$\Rightarrow \langle K \rangle = \frac{3}{2} N k_B T$$

"INSTANTANEOUS TEMPERATURE"

$$K(t) = \frac{3}{2} N k_B T(t)$$

MEAN SQUARE DISPLACEMENT (AND DIFFUSION)

$$\left\langle |\vec{r}(t) - \vec{r}(0)|^2 \right\rangle$$

$$D = \lim_{t \to \infty} \frac{1}{6t} \left\langle |\vec{r}(t) - \vec{r}(0)|^2 \right\rangle$$

Diffusion

Assume uncorrelated "jumps"

τ = mean time between jumps

a = jump distance

t ($\gg\tau$) = observation time

$$\langle |\Delta\vec{r}|^2 \rangle = \left\langle \left| \sum_i \vec{r}_i \right|^2 \right\rangle =$$

$$= \left\langle \sum_{ij} \vec{r}_i \cdot \vec{r}_j \right\rangle =$$

$$= \left\langle \sum_i r_i^2 \right\rangle =$$

$$= \sum_i \langle r_i^2 \rangle = \frac{t}{\tau} a^2 = 6Dt$$

Pressure

Clausius virial function: $W(\{\vec{r}_i\}) = \sum_1^N \vec{r}_i \cdot \vec{F}_i^{TOT}$

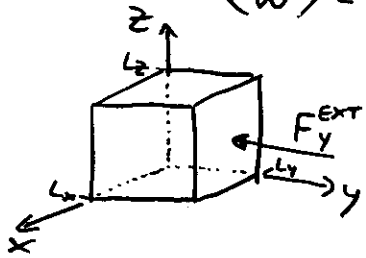
$$\langle W \rangle = \lim_{t\to\infty} \frac{1}{t} \int_0^t d\tau \underbrace{\sum_1^N \vec{r}_i(\tau) \cdot m_i \ddot{\vec{r}}_i(\tau)}_{\dot{W}} =$$

$$= -\lim_{t\to\infty} \frac{1}{t} \int_0^t d\tau \sum_1^N m_i |\dot{\vec{r}}_i(\tau)|^2 =$$

$$= -DNk_BT \qquad \text{[by equipartition]}$$

$$\vec{F}_i^{TOT} = \vec{F}_i + \vec{F}_i^{EXT}$$

$$\langle W \rangle = \langle W^{INT} \rangle + \langle W^{EXT} \rangle$$



$$\langle W^{EXT} \rangle = -DPV$$

$$PV = Nk_BT + \frac{1}{D} \left\langle \sum_1^N \vec{r}_i \cdot \vec{F}_i \right\rangle$$

The virial equation

Case of pairwise interactions

$$W^{INT} \equiv \sum_{1}^{N} \vec{r}_i \cdot \vec{F}_i \qquad \vec{F}_i = \sum_{j \neq i} \vec{f}_{ij}$$

$$= \frac{1}{2} \sum_i \sum_{j \neq i} \left(\vec{r}_i \cdot \vec{f}_{ij} + \vec{r}_j \cdot \vec{f}_{ji} \right) \qquad \vec{f}_{ji} = -\vec{f}_{ij}$$

$$= \frac{1}{2} \sum_i \sum_{j \neq i} \vec{r}_{ij} \cdot \vec{f}_{ij}$$

$$= \sum_i \sum_{j > i} \vec{r}_{ij} \cdot \vec{f}_{ij} \qquad \Leftarrow \text{PBC-compatible}$$

But $\vec{f}_{ij} = -\left.\frac{d\phi}{dr}\right|_{r_{ij}} \frac{\vec{r}_{ij}}{r_{ij}}$

$$W^{INT} = -\sum_i \sum_{j>i} r_{ij} \left.\frac{d\phi}{dr}\right|_{r_{ij}}$$

$$PV = N k_B T - \frac{1}{D} \left\langle \sum_i \sum_{j>i} r_{ij} \left.\frac{d\phi}{dr}\right|_{r_{ij}} \right\rangle$$

Starting a simulation

- From scratch
  - * Assign $\vec{r}_i(0)$, $\vec{v}_i(0)$. Popular choices:
    - $\vec{r}_i$ lattice + <u>random</u> displacements
      $\vec{v}_i = 0$
    - $\vec{r}_i$ lattice (perfect)
      $\vec{v}_i$ taken from Maxwellian (<u>random</u>)
  - * Start run, system will reach equilibrium
- From previous run
  - * $\vec{r}_i, \vec{v}_i$ from last step of previous run
  - * Already equilibrated, but now user may have changed parameters ($T, P, \ldots$)
    $\Rightarrow$ new equilibration

Equilibration

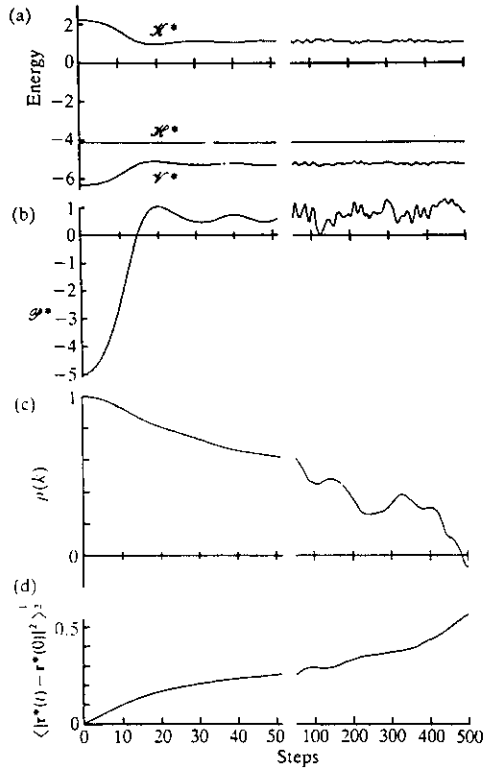
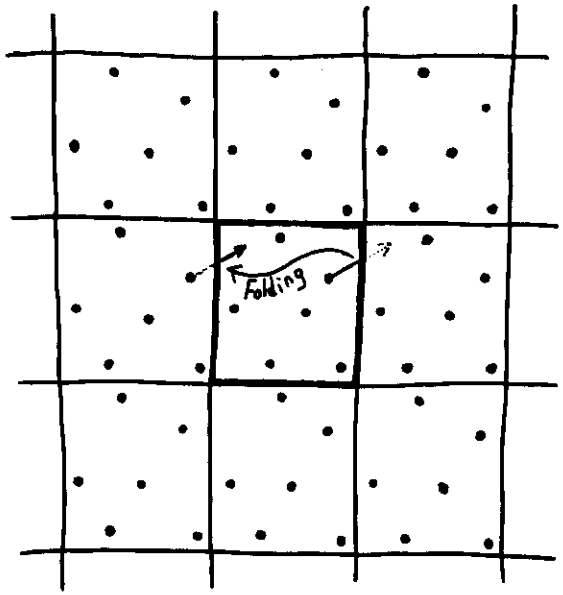


Fig. 5.11 The equilibration phase of a MD simulation. The first 50 steps are shown in detail. The system consists of 108 atoms interacting via the Lennard-Jones pair potential, starting from an f.c.c. lattice with Maxwell–Boltzman velocity distribution. The system is near the triple point ($\rho^* = 0.8442$, $T^* = 0.722$, $\delta t^* = 0.005$, $r_c^* = 2.5$, no long-range corrections applied). (a) Potential, kinetic, and total energies; (b) instantaneous pressure; (c) translational order parameter; (d) root-mean-square displacement.

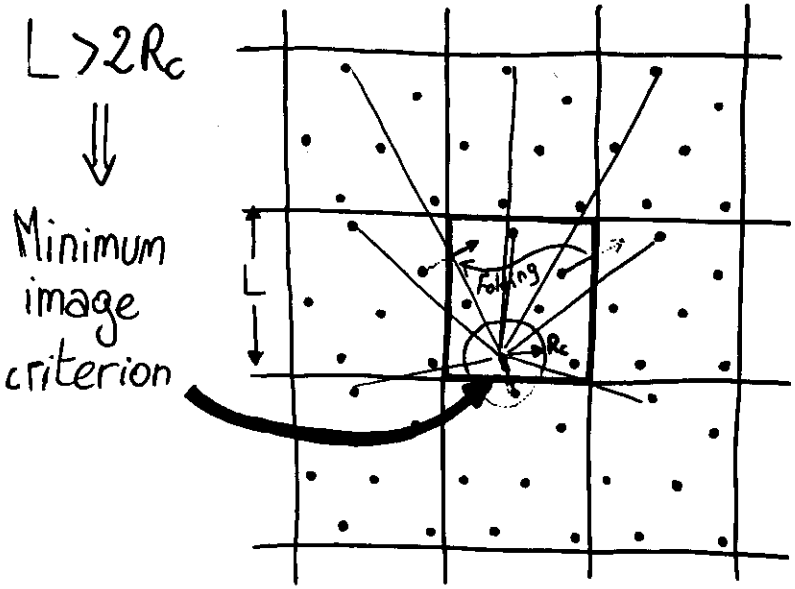
Periodic boundary conditions

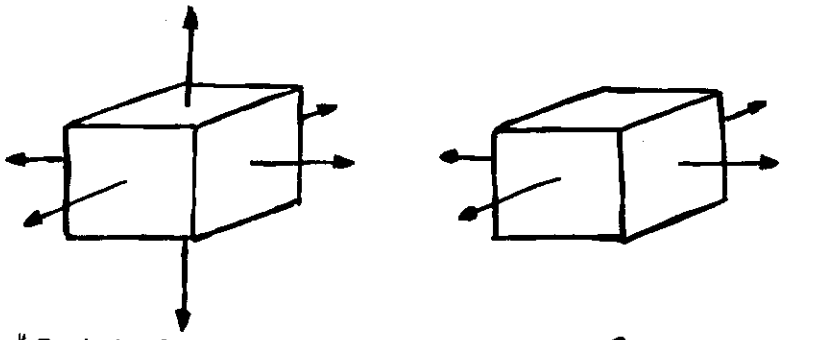
- Elimination of surface effects
- "Infinite systems"



Periodic boundary conditions

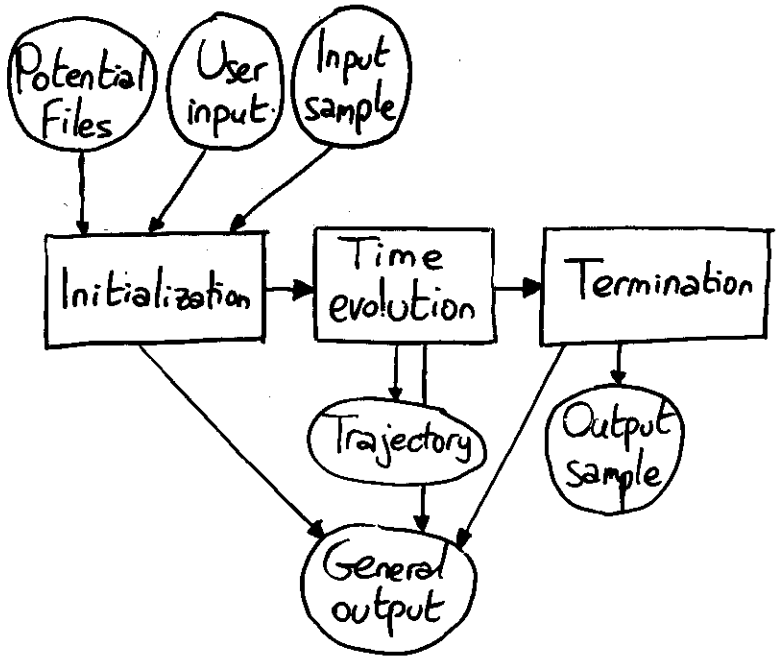
- Elimination of surface effects
- "Infinite systems"

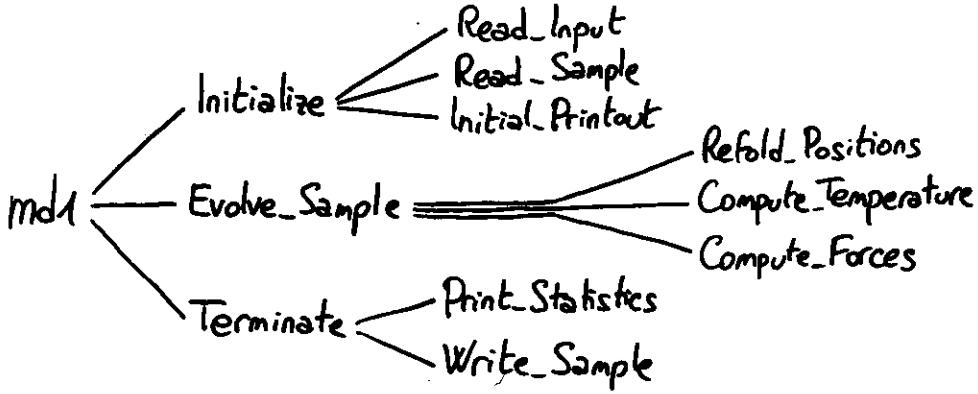




"Bulk" (no surfaces) "Slab" (two surfaces)

General structure of MD codes





```
program MD1
!
!  This is the main driver for the molecular dynamics program.
!
implicit none
call Initialize
call Evolve_Sample
call Terminate
end program MD1
```

```
module Particles
!
!  Contains all the data structures containing informations
!  about atom properties
!
   integer, parameter :: DIM=3      ! change DIM to switch between 2D and 3D !
   logical :: VelAcc = .FALSE.      ! velocities and accelerations in file?
   integer :: N=0
   double precision, dimension(DIM) :: BoxSize
!     the following arrays are allocated at run time,
!     when the number of atoms N is known.
   double precision, dimension(:,:), allocatable :: pos     ! positions
   double precision, dimension(:,:), allocatable :: vel     ! velocities
   double precision, dimension(:,:), allocatable :: acc     ! accelerations
   double precision, dimension(:), allocatable :: ene_pot   ! potential energies
   double precision, dimension(:), allocatable :: ene_kin   ! kinetic energies
!     scalars derived from the above quantities
   double precision :: volume, density                    ! these are constants
   double precision :: virial       ! virial term, used to compute the pressure
end module Particles
```

```
subroutine Initialize
!
!  Initialization procedure (called once at the beginning, before the
!  time evolution starts)
!
use Particles
implicit none
!
!  Read the user directives controlling the simulation
!
call Read_Input
!
!  Read the input sample containing the initial particle coordinates
!  (and perhaps also velocities and accelerations)
!
call Read_Sample
!
!  Print informations on the run on the standard output file
!
call Initial_Printout
end subroutine Initialize
```

```
subroutine Read_Sample
                              ( DECLARATIONS OMITTED )
read(1,*,end=600,err=900) VelAcc,N,BoxSize
volume  = product(BoxSize)
density = N / volume
if (ChangeRho) then
   scale = ( density / RhoRequested ) ** ( 1.d0 / DIM )
   BoxSize = scale*BoxSize
   volume  = product(BoxSize)
   density = N / volume
endif

allocate( pos(DIM,N) )
allocate( vel(DIM,N) )
allocate( acc(DIM,N) )
allocate( ene_pot(N) )
allocate( ene_kin(N) )

do i=1,N
   read(1,*,end=800,err=900) PosAtomReal
   pos(:,i) = PosAtomReal/BoxSize
enddo
         ( ALSO READ VELOCITIES AND ACCELERATIONS IF PRESENT )

Mass_Center = sum( pos , dim=2 ) / N
do k=1,DIM
   pos(k,:) = pos(k,:) - Mass_Center(k)
enddo
                ( INPUT/OUTPUT ERROR HANDLING OMITTED )

end subroutine Read_Sample
```

```
subroutine Terminate
!
!  Termination procedure (called once at the end, after time evolution
!  and before program termination)
!
use Particles
implicit none
!
!  Print a line with averages, etc
!
call Print_Statistics
!
!  Write the final sample (positions, velocities, accelerations) on file
!
call Write_Sample
!
!  Deallocate all the dynamical arrays to clean memory ('ecological' practice)
!
deallocate( pos )
deallocate( vel )
deallocate( acc )
deallocate( ene_pot )
deallocate( ene_kin )
end subroutine Terminate
```

```
subroutine Evolve_Sample

!  This is the main subroutine, controlling the time evolution of the system.

                              ( DECLARATIONS OMITTED )

call Compute_Temperature(ene_kin_aver,temperature)
time: do step=1,Nsteps
   call Refold_Positions
   pos = pos + deltat*vel + 0.5d0*(deltat**2)*acc       ! r(t+dt)
   if (ConstantT .and. (temperature > 0) ) then   !  veloc rescale for const T
      call Compute_Temperature(ene_kin_aver,temperature) ! T(t)
      chi = sqrt( Trequested / temperature )
      vel = chi*vel + 0.5d0*deltat*acc                  ! v(t+dt/2)
   else                                !  regular constant E dynamics
      vel = vel + 0.5d0*deltat*acc                      ! v(t+dt/2)
   endif
   call Compute_Forces                                  ! a(t+dt),ene_pot,virial
   vel = vel + 0.5d0*deltat*acc                         ! v(t+dt)
   call Compute_Temperature(ene_kin_aver,temperature)   ! at t+dt, also ene_kin
   ene_pot_aver = sum( ene_pot ) / N
   ene_tot_aver = ene_kin_aver + ene_pot_aver
   pressure = density*temperature + virial/volume
   print '(1x,i6,5f14.6)',step,temperature, &
                           ene_kin_aver,ene_pot_aver,ene_tot_aver,pressure
!     accumulate statistics:
   temperature_sum = temperature_sum + temperature
   ene_kin_sum     = ene_kin_sum     + ene_kin_aver
   ene_pot_sum     = ene_pot_sum     + ene_pot_aver
   pressure_sum    = pressure_sum    + pressure
enddo time
end subroutine Evolve_Sample
```

```
subroutine Refold_Positions
!
!  Particles that left the box are refolded back into the box by
!  periodic boundary conditions
!
use Particles
implicit none
where ( pos >  0.5d0 ) pos = pos - 1.d0
where ( pos < -0.5d0 ) pos = pos + 1.d0
end subroutine Refold_Positions
```

```
subroutine Compute_Temperature(ene_kin_aver,temperature)
!
!  Starting from the velocities currently stored in vel, it updates
!  the kinetic energy array ene_kin, and computes and returns the
!  averaged (on particles) kinetic energy ene_kin_aver and the
!  instantaneous temperature.
!
use Particles
implicit none
double precision :: ene_kin_aver, temperature
double precision, dimension(DIM) :: real_vel
integer :: i
do i=1,N
   real_vel = BoxSize*vel(:,i)                          ! real space velocity of i
   ene_kin(i) = 0.5d0*dot_product(real_vel,real_vel)    ! kin en of each atom
enddo
ene_kin_aver = sum( ene_kin ) / N
temperature = 2.d0*ene_kin_aver/DIM
end subroutine Compute_Temperature
```

```
subroutine Compute_Forces
                              ( DECLARATIONS OMITTED )

ene_pot = 0.d0
acc = 0.d0
virial = 0.d0
do i = 1,N-1                                          ! looping an all pairs
   do j = i+1,N
      Sij = pos(:,i) - pos(:,j)                       ! distance vector between i j
      where ( abs(Sij) > 0.5d0 )                      ! (in box scaled units)
         Sij = Sij - sign(1.d0,Sij)                   ! periodic boundary conditions
      end where                                       ! applied where needed.
      Rij = BoxSize*Sij                               ! go to real space units
      Rsqij = dot_product(Rij,Rij)                    ! compute square distance
      if ( Rsqij < Rcutoff**2 ) then                  ! particles are interacting?
                                                      ! compute Lennard-Jones potntl
         rm2 = 1.d0/Rsqij                             !  1/r^2
         rm6 = rm2**3                                 !  1/r^6
         rm12 = rm6**2                                !  1/r^12
         phi  = 4.d0 * ( rm12 - rm6 ) - phicutoff     !  4[1/r^12 - 1/r^6] - phi(Rc)
                                          !  The following is dphi = -(1/r)(dV/dr)
         dphi = 24.d0*rm2*( 2.d0*rm12 - rm6 )         !  24[2/r^14 - 1/r^8]
         ene_pot(i) = ene_pot(i) + 0.5d0*phi          ! accumulate energy
         ene_pot(j) = ene_pot(j) + 0.5d0*phi          ! (i and j share it)
         virial = virial - dphi*Rsqij                 ! accumul. virial=sum r(dV/dr)
         acc(:,i) = acc(:,i) + dphi*Sij               ! accumulate forces
         acc(:,j) = acc(:,j) - dphi*Sij               !    (Fji = -Fij)
       endif
    enddo
enddo
virial = - virial/DIM                                 ! definition of virial term
end subroutine Compute_Forces
```

md1 :seven lines of input

1) Arbitrary title string
2) Name of File with input sample
3) Name of file to contain output sample
4) Number of time steps
5) Amplitude of time step
6) Density (0 = leave unchanged)
7) Temperature (-1 = run at constant E)

Some possible problems with crystal/md1: select one.

PROBLEMS

RUNNING

- Study the behavior of trajectories when the time step is changed.
  A good reference value for the LJ system could be 0.0032

- Study the behavior of the caloric curve E(T) and of the pressure P(T)/kT by means of constant energy runs at a fixed density, starting from a crystalline arrangement.
  Select a density between 0.6 and 0.8 and go up until T=1.5.
  Watch for a solid-liquid transition.

- Study the behavior of E and P as a function of density for a fixed temperature, for instance T=1.15.
  Go up to density=1.

CODING

- Insert calculations of the total linear and angular momenta into MD1, and check their conservation.

- Insert calculation of the mean square displacement (between the beginning of the run and the current time steps).

- Remove periodic boundary conditions in order to simulate a cluster. Start from a cubic cluster with 32 atoms (2 cells per side in the input of crystal). Heat it up by means of a chain of runs with a temperature step of 0.05, and discuss the behavior of E(T).

Verlet algorithm implementations

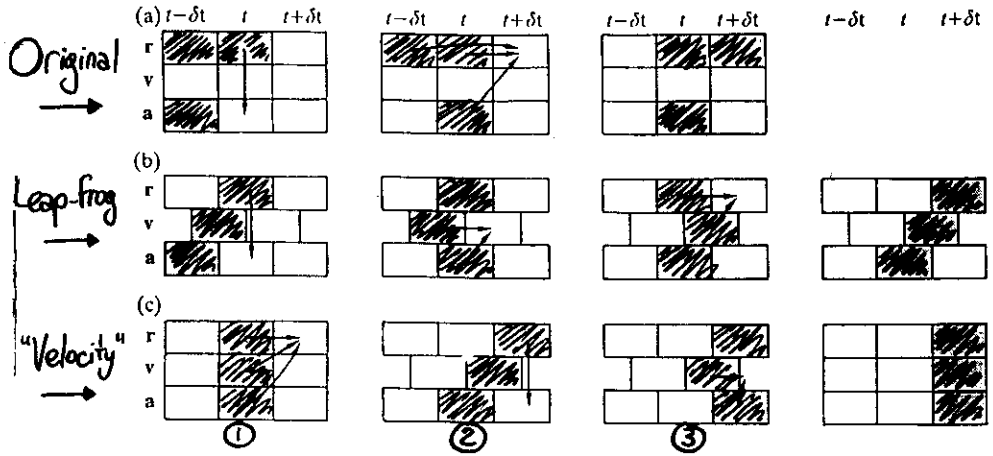


Fig. 3.2 Various forms of the Verlet algorithm. (a) Verlet's original method. (b) The leap-frog form. (c) The velocity form. We show successive steps in the implementation of each algorithm. In each case, the stored variables are in grey boxes.

Verlet: $\vec{r}(t+\Delta t) = 2\vec{r}(t) - \vec{r}(t-\Delta t) + \vec{a}(t)\Delta t^2$

"Velocity" implementation:

$$①\begin{cases} \vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 \\ \vec{v}(t+\frac{\Delta t}{2}) = \vec{v}(t) + \frac{1}{2}\vec{a}(t)\Delta t \end{cases}$$

$$② \quad \vec{a}(t+\Delta t) = -\frac{1}{m}\vec{\nabla}V(\vec{r}(t+\Delta t))$$

$$③ \quad \vec{v}(t+\Delta t) = \vec{v}(t+\frac{\Delta t}{2}) + \frac{1}{2}\vec{a}(t+\Delta t)\Delta t$$

Verlet algorithm implementations

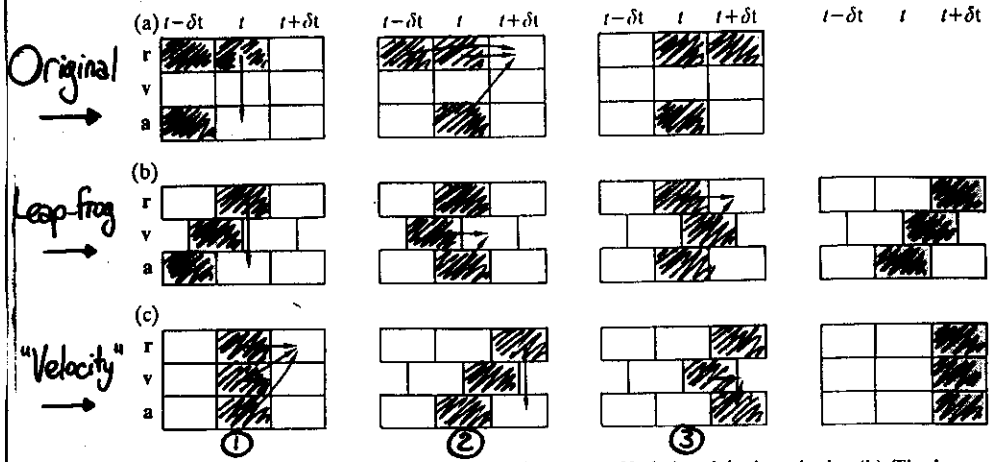


Fig. 3.2 Various forms of the Verlet algorithm. (a) Verlet's original method. (b) The leap-frog form. (c) The velocity form. We show successive steps in the implementation of each algorithm. In each case, the stored variables are in grey boxes.

Verlet: $\vec{r}(t+\Delta t) = 2\vec{r}(t) - \vec{r}(t-\Delta t) + \vec{a}(t)\Delta t^2$

"Velocity" implementation:

$$①\begin{cases} \vec{r}(t+\Delta t) = \vec{r}(t) + \vec{v}(t)\Delta t + \frac{1}{2}\vec{a}(t)\Delta t^2 \\ \vec{v}(t+\frac{\Delta t}{2}) = \vec{v}(t) + \frac{1}{2}\vec{a}(t)\Delta t \end{cases}$$

$$\left[\text{OR } \vec{v}(t+\frac{\Delta t}{2}) = \sqrt{\frac{T_0}{T(t)}}\,\vec{v}(t) + \frac{1}{2}\vec{a}(t)\Delta t \quad \text{if } \frac{\text{I want}}{T=T_0}\right]$$

$$② \quad \vec{a}(t+\Delta t) = -\frac{1}{m}\vec{\nabla}V(\vec{r}(t+\Delta t))$$

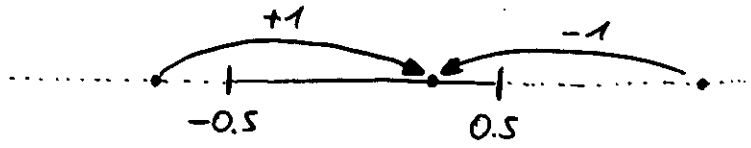
$$③ \quad \vec{v}(t+\Delta t) = \vec{v}(t+\frac{\Delta t}{2}) + \frac{1}{2}\vec{a}(t+\Delta t)\Delta t$$

Lot of time spent in

```
do i = ...
   do j = ...
      compute r_ij^2
      if r_ij^2 < R_c^2 then
      :
      endif
   enddo
enddo
```

most tests fail →

How to implement periodic boundary conditions?



There are many ways. Best is computer dependent

① IF $(X_{ij} > 0.5)$ $X_{ij} = X_{ij} - 1$
IF $(X_{ij} < -0.5)$ $X_{ij} = X_{ij} + 1$

(1bis) IF..THEN..
ELSE IF..
ENDIF

② IF $(|X_{ij}| > 0.5)$ $X_{ij} = X_{ij} - SIGN(1, X_{ij})$

③ $X_{ij} = X_{ij} - Int(X_{ij} + X_{ij})$

④ $X_{ij} = X_{ij} - Nint(X_{ij})$

① IBM mainframes

② IBM RS6000, SGI

③ Cray, Convex, HP 9000

④ DEC Alpha, DEC VAX

"IEEE craziness method"

[James Shearer, IBM Yorktown, private communication (1993)]

$$X_{ij} = X_{ij} - \left((X_{ij} + A_{magic}) - A_{magic} \right)$$

where

$$A_{magic} = 6755399441055744.D0$$

DISCLAIMER
I make no warranty of any kind with regard to the material shown in this transparency, including, but not limited to, fitness for any particular purpose related with atomistic computer simulation.

Avoid $\sqrt{\ }$ in the double loops!

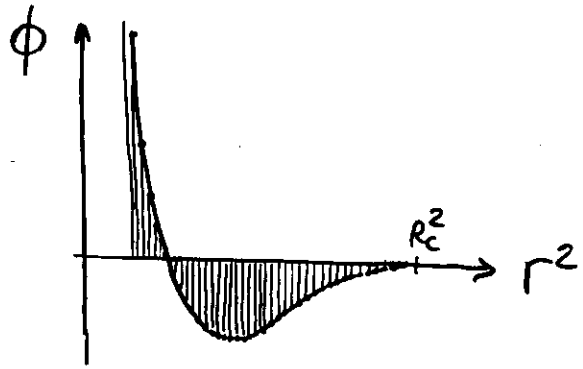
Try to do everything with r_{ij}^2 if you can.
In particular:

- always if $r_{ij}^2 < R_c^2$ then ...
- never $r_{ij} = \mathrm{sqrt}(r_{ij}^2)$
  if $r_{ij} < R_c$ then ...

For LJ, r_{ij} never needed
(see Compute_Forces in md1)

Other potentials may require it, but one can always use table look-ups

Table look-ups - The idea



- Set up Fine grid at constant Δr^2
- Sample once For all $\phi(r)$ and $\frac{1}{r}\frac{d\phi}{dr}$ over that grid — store samples in tables
- When you need potential and Forces For a certain r^2, do table look-up and interpolation

Done in md2

Table look-ups - implementation

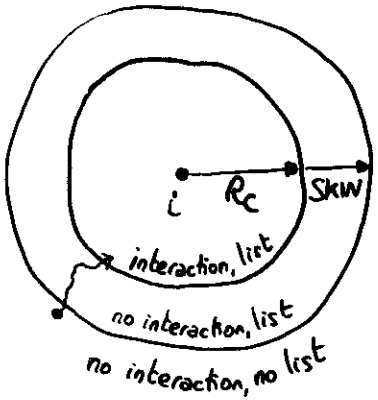
GENERATION

```
do k=1, TableSize
    Rsq = RsqMin + (k-1)*DeltaRsq
    (compute φ, 1/r dφ/dr => phi,dphi)
    PhiTab(k) = phi
    DPhiTab(k) = dphi
enddo
```

UTILIZATION

```
:
if (Rsqij < Rc^2) then
    rk = (Rsqij-RsqMin)*InvDeltaRsq+1
    k = int(rk)
              [between k and k+1]
    weight = rk-k
    phi = weight*PhiTab(k+1)
       +(1-weight)*PhiTab(k)
    dphi = (..same..)
:
```

Verlet neighbor list - The idea



- For each i construct list $\{j : R_{ij} < R_c + SKIN\}$
- More atoms than needed at present time

BUT:

List used For a while

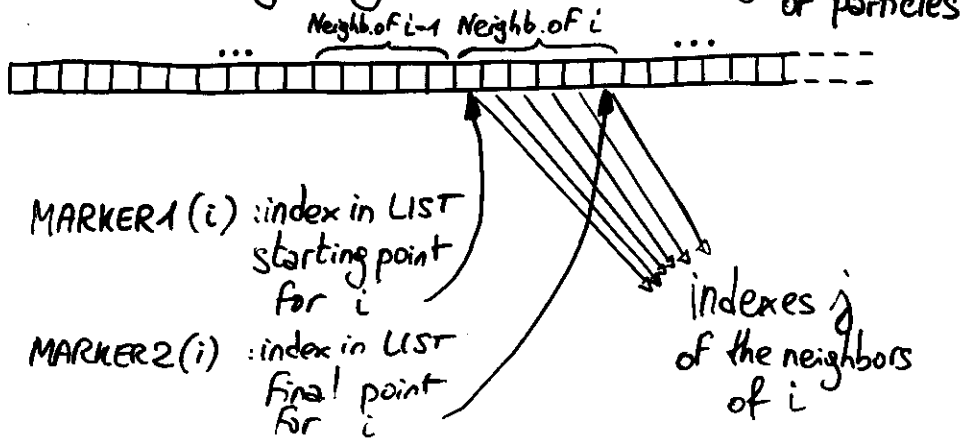
At the time steps done with the "old" list:

$\Rightarrow$ fixed volume $\Rightarrow O(N)$

List Formation still $O(N^2)$

Verlet neighbor list - implementation

"LIST" = A long integer vector containing indexes of particles



Once made, used in this way: [O(N)]

```
do i = 1, N
   do L = Marker1(i), Marker2(i)
      j = LIST(L)
      Compute rij
      if rij < Rc perform action
   enddo
enddo
```

In md3

```
subroutine Compute_Forces
                              (DECLARATIONS OMITTED)
ene_pot = 0.d0
acc = 0.d0
virial = 0.d0
!
!                Loop over particles in the neighbor list
do i = 1,N
   do L = Marker1(i),Marker2(i)                          ! part of List useful for i
      j = List(L)                                        ! take neigh index out of List
      Sij = pos(:,i) - pos(:,j)                          ! distance vector between i j
      where ( abs(Sij) > 0.5d0 )                         ! (in box scaled units)
         Sij = Sij - sign(1.d0,Sij)                      ! periodic boundary conditions
      end where                                          ! applied where needed.
      Rij = BoxSize*Sij                                  ! go to real space units
      Rsqij = dot_product(Rij,Rij)                       ! compute square distance
      if ( Rsqij < Rcutoff**2 ) then                     ! particles are interacting?
         rk = (Rsqij - RsqMin)*InvDeltaRsq + 1.d0        ! "continuous" index in table
         k = int(rk)                                     ! discretized index
         if (k < 1) k = 1                                ! unlikely but just to protect
         weight = rk - k                                 ! fractional part, in [0,1]
         phi  = weight* PhiTab(k+1) + (1.d0-weight)* PhiTab(k)  ! do linear
         dphi = weight*DPhiTab(k+1) + (1.d0-weight)*DPhiTab(k)  ! interpolation
         ene_pot(i) = ene_pot(i) + 0.5d0*phi             ! accumulate energy
         ene_pot(j) = ene_pot(j) + 0.5d0*phi             ! (i and j share it)
         virial = virial - dphi*Rsqij                    ! accumul. virial=sum r(dV/dr)
         acc(:,i) = acc(:,i) + dphi*Sij                  ! accumulate forces
         acc(:,j) = acc(:,j) - dphi*Sij                  !    (Fji = -Fij)
      endif
   enddo
enddo
virial = - virial/DIM                                    ! definition of virial term
end subroutine Compute_Forces
```

Construction of a neighbor list

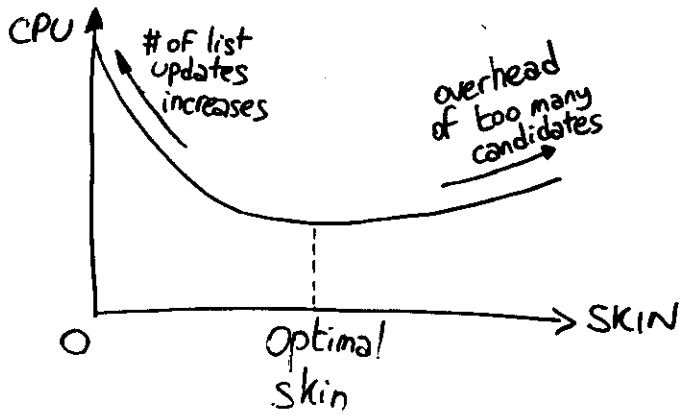
$O(N^2)$

```
L = 1
do i = 1, N
   Marker1(i) = L
   do j = i+1, N
      compute r_ij
      if r_ij < R_c + SKIN then
         LIST(L) = j
         L = L + 1
      endif
   enddo
   Marker2(i) = L - 1
enddo
```

In md3

```
subroutine Update_List(Range)
                                   (DECLARATIONS OMITTED)
RangeSq = Range**2
!
!  The following is the Fincham-Ralston loop for list updating
!
L = 1                                                  ! initialize index into List
do i = 1,N                                             ! looping an all pairs
   do j = i+1,N
      Sij = pos(:,i) - pos(:,j)                        ! distance vector between i j
      where ( abs(Sij) > 0.5d0 )                       ! (in box scaled units)
         Sij = Sij - sign(1.d0,Sij)                    ! periodic boundary conditions
      end where                                        ! applied where needed.
      Rij = BoxSize*Sij                                ! go to real space units
      Rsqij = dot_product(Rij,Rij)                     ! compute square distance
      if ( Rsqij < RangeSq ) then                      ! is j to be a neighbor of i?
         Advance(j) = 1                                !    if yes, put a 1,
      else                                             !    otherwise
         Advance(j) = 0                                !    put a 0, in Advance
      endif
   enddo
   Marker1(i) = L                                      ! list for i starts here
   do j = i+1,N
      if ( L > MaxListLength) goto 9999                ! panic exit if no more room
      List(L) = j                                      ! j will be really included ..
      L = L + Advance(j)                               ! only of Advance(j) is 1
   enddo
   Marker2(i) = L - 1                                  ! list for i terminates here
enddo
ListLength = L - 1                                     ! final used length of List
write(6,'(i8,a)') ListLength,' indexes in list'        ! complete report line
DisplaceList = 0.d0                                    ! reset displacement vectors
end subroutine Update_List
```

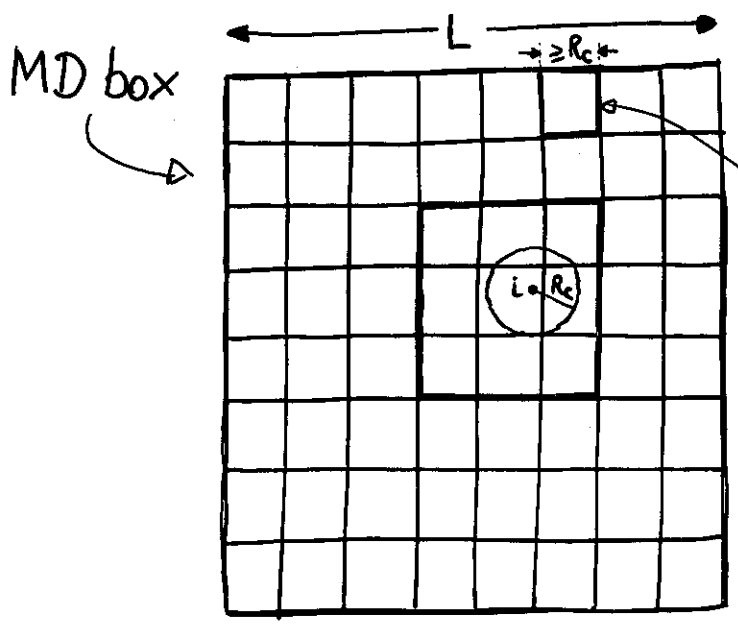
Optimizing your skin



- Trajectories remain unchanged
- Optimal skin depend on thermodynamic state – Increases with T

Try this with md3

Cell method - The idea



Assume $L \gg R_c$

Particles candidate to interact with i are certain located in

- the same cell of i

OR

- one of the neighboring cells [8 if 2D, 26 if 3D]

⇒ Fixed volume (independent from L)
⇒ O(N) method

Cell method - sketch of implementation

- Establish number of cells along x, y, z
  $N_x = int(L_x/R_c)$, $N_y = int(L_y/R_c)$, $N_z = int(L_z/R_c)$
  $M = N_x N_y N_z$ cells $\quad c = 1, \ldots, M$
- Construct $C_i : i \in C_i$ for each i
- Construct a "cell contents" data structure, inverse of the above:

$$C \rightarrow \{i_1, i_2 \ldots, i_{N_C} : C_{i_\ell} = C \quad (\ell = 1, \ldots, N_C)\}$$

Two ways to do this
- linked list
- two-index array ←
  contents(1-N_C, C)
  (containing indexes)

```
do c = 1, ..., M
  {cand_other} = {indexes of particles in c' neighbor of c}
  do ℓ = 1, ... N_C
    i = contents(ℓ, c)
    {cand_same} = {indexes of particles in c other than i}
    do j ∈ {cand_other} + {cand_same}
      compute r_ij
      if r_ij < R_c perform action on i and j
    enddo
  enddo
enddo
```

V.I. Arnold and A. Avez,
"Ergodic problems of classical mechanics",
Benjamin, 1968.

CHAPTER 1

DYNAMICAL SYSTEMS

This chapter contains examples of dynamical systems and related problems.

§1. Classical Systems

DEFINITION 1.1

Let M be a smooth manifold, μ a measure on M defined by a continuous positive density, ϕ_t: $M \to M$ a one-parameter group of measure-preserving diffeomorphisms. The collection (M, μ, ϕ_t) is called a classical dynamical system.

The parameter t is a real number or an integer. If $t \in \mathbf{R}$, the group ϕ_t is usually defined in local coordinates by:

$$\dot{x}^i = f^i(x^1, \ldots, x^n), \quad i = 1, \ldots, \quad n = \dim M.$$

If $t \in \mathbf{Z}$, ϕ_t is the discrete group generated by a measure-preserving diffeomorphism $\phi = \phi_1$. Then the system is merely denoted by (M, μ, ϕ) and ϕ is called the automorphism.

EXAMPLE 1.2. QUASI-PERIODIC MOTION

Let M be the torus $\{(x, y) \bmod 1\}$. The measure is $dx\,dy$, the group ϕ_t is a translation group:

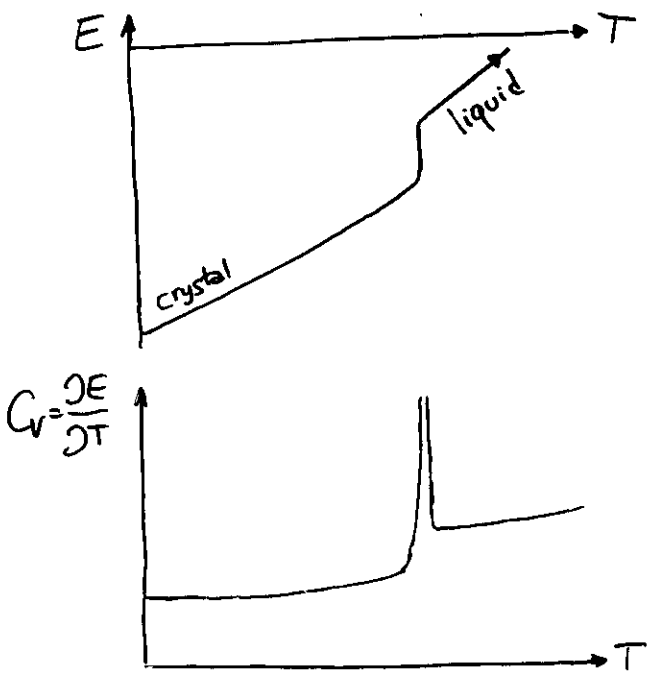
$$\dot{x} = 1, \quad \dot{y} = \alpha$$

where $\alpha \in \mathbf{R}$, and dot denotes d/dt. Assume $\alpha = p/q$ rational:

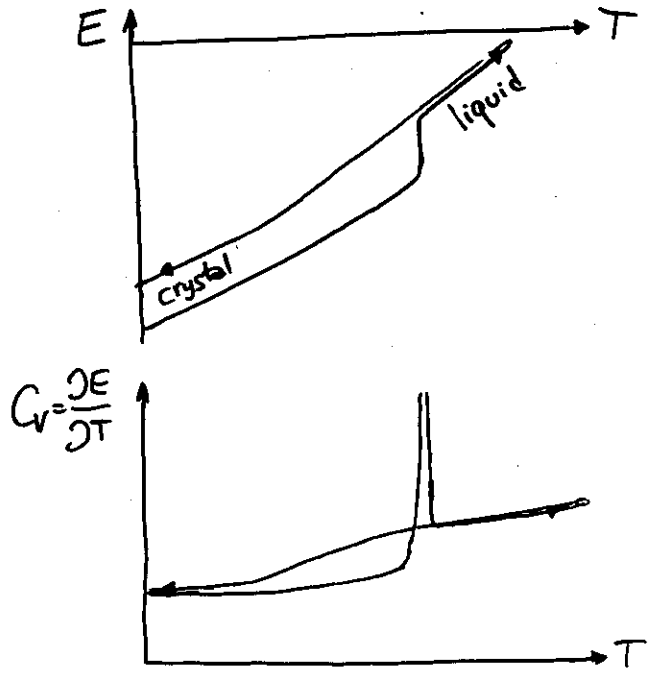
$$p, q \in \mathbf{Z}, \quad q > 0$$

and p and q relatively prime. In the covering plane (x, y), the orbit with

Caloric curve



Caloric curve: plays tricks



Ensembles

$$\langle A \rangle = \frac{1}{t_{obs}} \sum_{1}^{t_{obs}} A(\Gamma(t))$$

$$\Gamma(t) \equiv \{\vec{r}_1(t), \dots, \vec{r}_N(t), \vec{v}_1(t), \dots, \vec{v}_N(t)\}$$

Newton's equation $\Rightarrow$ sampling in the microcanonical ensemble

$$\rho_{NVE} \propto \delta(H(\Gamma) - E)$$

$\langle A \rangle$ above is really : $\langle A \rangle_{NVE}$

Goal :

Find other equation $\Rightarrow$ sampling in ρ_{ens}

$$\langle A \rangle_{ens} = \frac{1}{\mathcal{C}_{obs}} \sum_{1}^{\mathcal{C}_{obs}} A(\Gamma(\mathcal{C}))$$

$\mathcal{C}$: "time-like" variable

Isoenthalpic-isobaric ensemble (NPH)

$H = E + PV$, V varying , P Fixed

Andersen (1980)

$\vec{r}_i = V^{1/3} \vec{s}_i$
$\vec{v}_i = V^{1/3} \dot{\vec{s}}_i$

Introduce "piston" with mass Q : additional degree of freedom

$$\mathcal{L} = \underbrace{\frac{1}{2} V^{2/3} \sum_i m_i \dot{s}_i^2}_{\text{kinetic, particles}} + \underbrace{\frac{1}{2} Q \dot{V}^2}_{\text{kinetic piston}} - \underbrace{U(V^{1/3} \vec{s}_i)}_{\text{potential, particle}} - \underbrace{PV}_{\text{potential, piston}}$$

$$\begin{cases} \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{s}_i} - \frac{\partial \mathcal{L}}{\partial s_i} = 0 \\ \frac{d}{dt} \frac{\partial \mathcal{L}}{\partial \dot{V}} - \frac{\partial \mathcal{L}}{\partial V} = 0 \end{cases} \Rightarrow \begin{cases} \ddot{\vec{s}}_i = \frac{1}{m_i V^{1/3}} \vec{F}_i - \frac{2}{3} \dot{\vec{s}}_i \frac{\dot{V}}{V} \\ \ddot{V} = \frac{1}{Q} [\mathcal{P} - P] \end{cases}$$

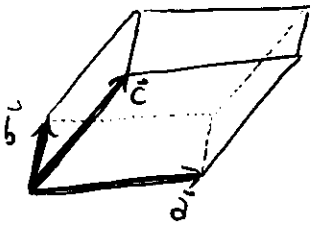
where $\mathcal{P} = \frac{N k_B T(t)}{V} + \frac{1}{3V} \sum_i \vec{r}_i \cdot \vec{F}_i$

NVE recovered when $Q \to \infty$

Q chosen on convenience basis

Extension : box with varying shape

Parrinello and Rahman (1980)



$$\underline{\underline{h}} = \left(\left(\vec{a}\right)\left(\vec{b}\right)\left(\vec{c}\right) \right) \qquad V = \det \underline{\underline{h}}$$

9 additional degrees of freedom

$$\vec{r}_i = \underline{\underline{h}}\, \vec{s}_i$$

$$\vec{p}_i = m_i \underline{\underline{h}}\, \dot{\vec{s}}_i$$

$$\mathcal{L} = \sum_i \frac{p_i^2}{2m_i} + \frac{1}{2} Q \sum_{\alpha\beta}^{3} \dot{h}_{\alpha\beta}^2 - U(\vec{r}_i, \ldots, \vec{r}_N)$$

$$\begin{cases} \dfrac{d}{dt}\dfrac{\partial \mathcal{L}}{\partial \dot{s}_i} - \dfrac{\partial \mathcal{L}}{\partial s_i} = 0 \\ \dfrac{d}{dt}\dfrac{\partial \mathcal{L}}{\partial \dot{h}_{\alpha\beta}} - \dfrac{\partial \mathcal{L}}{\partial h_{\alpha\beta}} = 0 \end{cases} \Rightarrow (\ldots)$$

M. Parrinello and A. Rahman

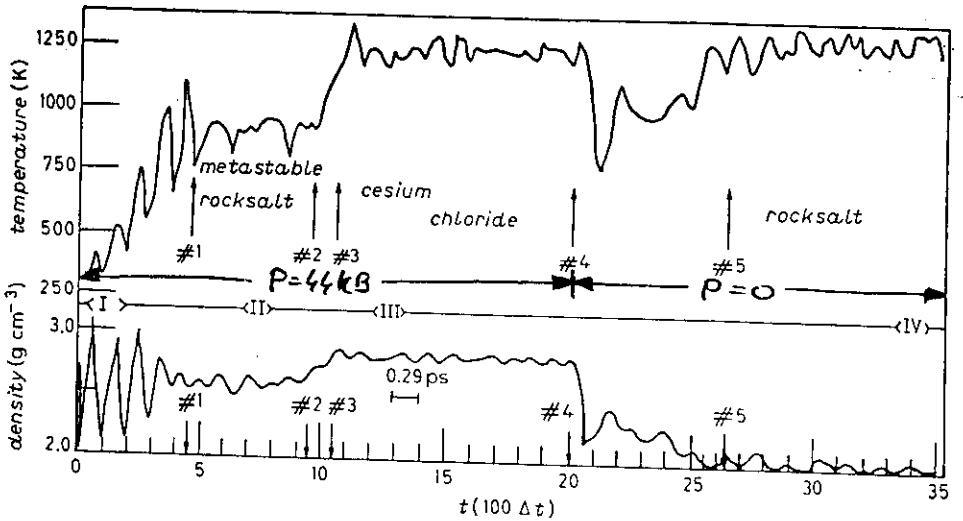
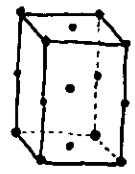
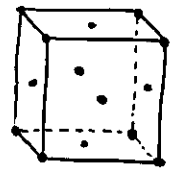


Fig. 2. – Time history of a compression and decompression molecular-dynamics run on KCl. Points plotted are 25 Δt apart, hence the nonsmooth appearance. In regions I to IV the pair correlations were monitored; these have not been shown in this lecture. Significance of the numbered vertical arrows is discussed in the text.



rocksalt structure



cesium chloride structure

Canonical ensemble (NVT)

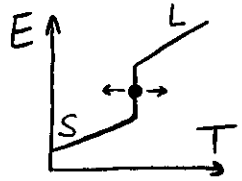
Nose` (1984) and Hoover (1985)

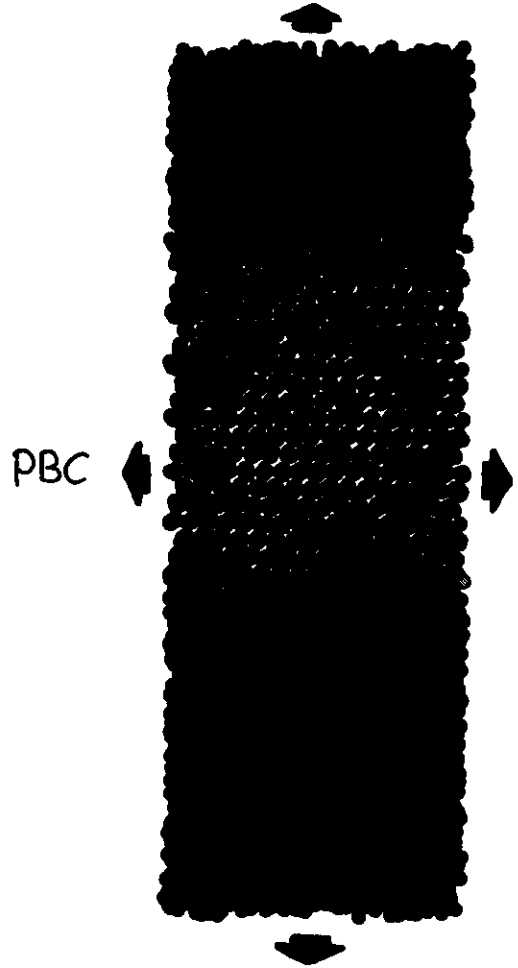
$$\dot{\vec{r}}_i = \frac{\vec{p}_i}{m}$$

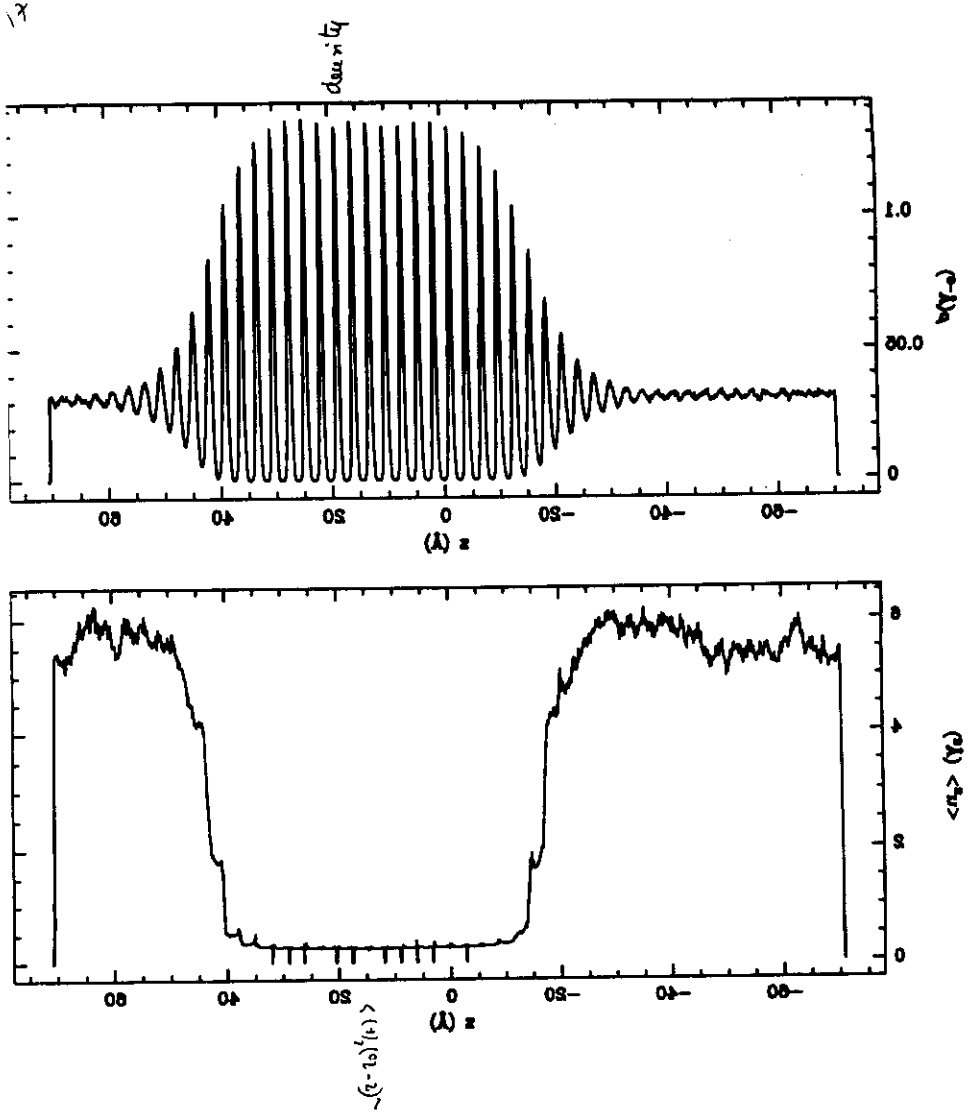
$$\dot{\vec{p}}_i = \vec{F}_i - \xi \vec{p}_i \qquad \text{time-dependent Friction}$$

$$\frac{Q}{2}\dot{\xi} = \sum_i \frac{p_i^2}{2m} - \frac{3N}{2} k_B T \qquad \text{Kinetic energy imbalance}$$

Determination of the melting point

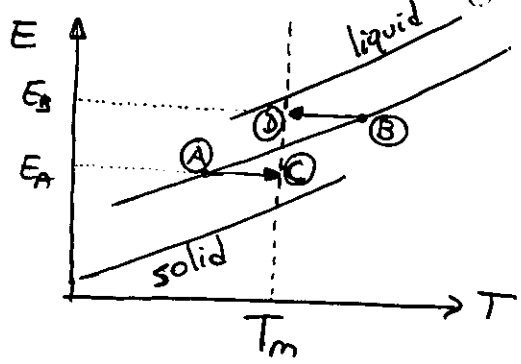






Procedure to determine T_m

- Construct system approximately half solid and half liquid
- Simulation will start from a point on the (unstable) green line

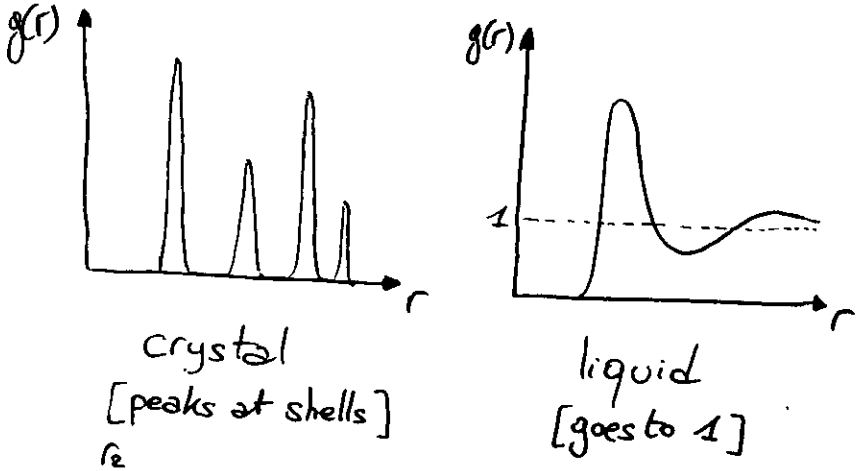


- Select an energy and start a NPH run
- IF E_A is selected, system will crystallize: Ⓐ→Ⓒ
- IF E_B is selected, system will melt: Ⓑ→Ⓓ
- In both cases, $T(t)-T_m \simeq [T(0)-T_m]e^{-t/\tau}$

Static structure: the pair correlation function g(r)

$g(r)$: probability to find a pair a distance r apart, relative to what expected for a random distribution at the same density

$$\rho g(r) = \frac{1}{N} \left\langle \sum_{1}^{N}{}_{i} \sum_{1 \; j \neq i}^{N} \delta(r - r_{ij}) \right\rangle$$

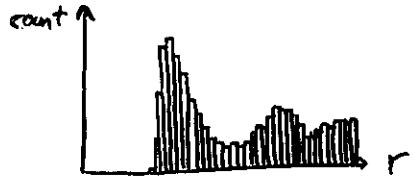


$\rho \int_{r_1}^{r_2} g(r) 4\pi r^2 dr$ = no. of atoms at a distance between r_1 and r_2

⇒ information on crystal structure

Calculation of g(r)

- Divide r axis in finite intervals
- Accumulate on histogram during the MD run
- Maximum possible distance is $\frac{L}{2}$




```
do i = 1, N-1
   do j = i+1, N
      (compute r_ij^2)
      r_ij = sqrt(r_ij^2)
      k = int(r_ij/Δr) + 1
      if (k ≤ k_max) then
         count(k) = count(k) + 1
      endif
   enddo
enddo
```

- Normalize at the end of the run
- Rather expensive $O(N^2)$ property
- Can do an histogram in r^2 to avoid the $\sqrt{}$

Static structure factor $S(k)$

$$\rho(\vec{k}) = \sum_{1}^{N} e^{i\vec{k}\cdot\vec{r}_i}$$

$$S(\vec{k}) \equiv \frac{1}{N}\left\langle \rho(\vec{k})\rho(-\vec{k})\right\rangle$$

$$= \frac{1}{N}\left\langle \sum_{ij} e^{i\vec{k}\cdot(\vec{r}_i-\vec{r}_j)}\right\rangle$$

$$= \frac{1}{N}\left\langle \sum_{ij} \cos(\vec{k}\cdot\vec{r}_{ij})\right\rangle$$

- Measured by scattering experiments
- Essentially a Fourier transform of $g(r)$
  $[S(k) = 1 + \rho\hat{g}(k)]$
- Result cannot depend on choice of image

$$e^{ik_x(x_i+L_x)} = e^{ik_x x_i}$$

$$e^{ik_x L_x} = 1 \quad \Rightarrow k_x = \frac{2\pi}{L_x} n_x$$

⇒ Allowed $\vec{k}$ are quantized as a consequence of using PBC

Dynamic structure:

space-time (Van Hove) correlation function

$G(\vec{r},t)$: ∝ probability that an atom is at position $\vec{r}$ at time t if there is an atom at position $\vec{0}$ at time 0

$$G(\vec{r},t) = G_{self}(\vec{r},t) + G_{distinct}(\vec{r},t)$$

$$S(\vec{k},\omega) = \iint e^{i(\vec{k}\cdot\vec{r}-\omega t)} G(\vec{r},t)\, d^3\vec{r}\, dt$$

self

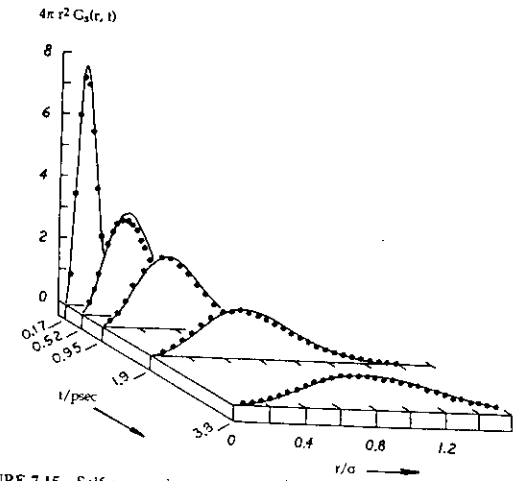


FIGURE 7.15 Self space–time correlation function for 256 Lennard-Jones atoms at $\rho^* = 0.75$, $T^* = 1.012$. Points are simulation results obtained from averages over 2000 time origins, each origin separated by 10 time-steps. Lines are from the Gaussian approximation (7.97) using values of the mean-square displacement obtained from the simulation.

distinct

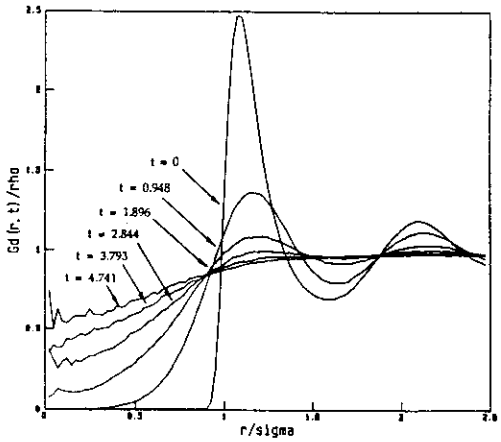


FIGURE 7.16 Distinct space–time correlation function $G_d(r,t)/\rho^*$ from a simulation of 256 Lennard-Jones atoms at $\rho^* = 0.75$, $T^* = 1.012$. Averages were computed over 2000 time origins, each origin separated by 10 time-steps. The delay times t are shown in picoseconds and were made dimensional using potential parameters characteristic of argon.

Phonon dispersions (with T dependence!)

- Do MD run and store trajectories $\{\vec{r}_i(t), \vec{v}_i(t)\}$
- Construct instantaneous density (or current) in reciprocal space for the desired $\vec{k}$ vectors:

$$\rho(\vec{k},t) = \sum_i e^{i\vec{k}\cdot\vec{r}_i(t)} \quad \left(\text{or } \vec{j}(\vec{k},t) = \sum_i \vec{v}_i(t) e^{i\vec{k}\cdot\vec{r}_i(t)}\right)$$

- Construct time-dependent density-density correlations ("intermediate scattering function"):

$$F(\vec{k},t) = \frac{1}{N}\left\langle \rho(\vec{k},t+t_0)\,\rho(-\vec{k},t_0)\right\rangle$$

[small $\vec{k}$ ⇒ long t]

- $$S(\vec{k},\omega) = \int dt\, e^{-i\omega t} F(\vec{k},t)$$

- Peaks in $(\vec{k},\omega)$ plane correspond to phonon dispersion.
- Broadening related to anharmonicity

J.P. Hansen and M.L. Klein (1976)

two different temperatures: $T^* = 0.735$ and $T^* = 1.20$; this phonon exhibits a sizable negative shift as T^* increases. Figure 7 shows the longitudinal phonon at $\vec{Q} = (2\pi/a)(\frac{1}{3}, 0, 0)$ under identical temperature and density conditions. This time the shift is small and positive. The apparent increase in intensity as T^* is increased is governed entirely by the sum rule equation (15).

E. Temperature shifts at constant pressure

Figure 8 shows the temperature shifts for transverse phonons propagating in [00ζ] direction for roughly the same $\rho^* - T^*$ states studied experimentally by Batchelder *et al.*[14] As expected the peaks shift to lower frequencies and broaden as T^* increases, and in addition the multiphonon background grows. This behavior is again compatible with the sum rule equation (15).

Figure 9 compares the MD percentage frequency shifts with the experimental results[14,29] and the results of a self-consistent-phonon calculation.[31] The MD results are in better agreement with the experimental data than the phonon calculations.

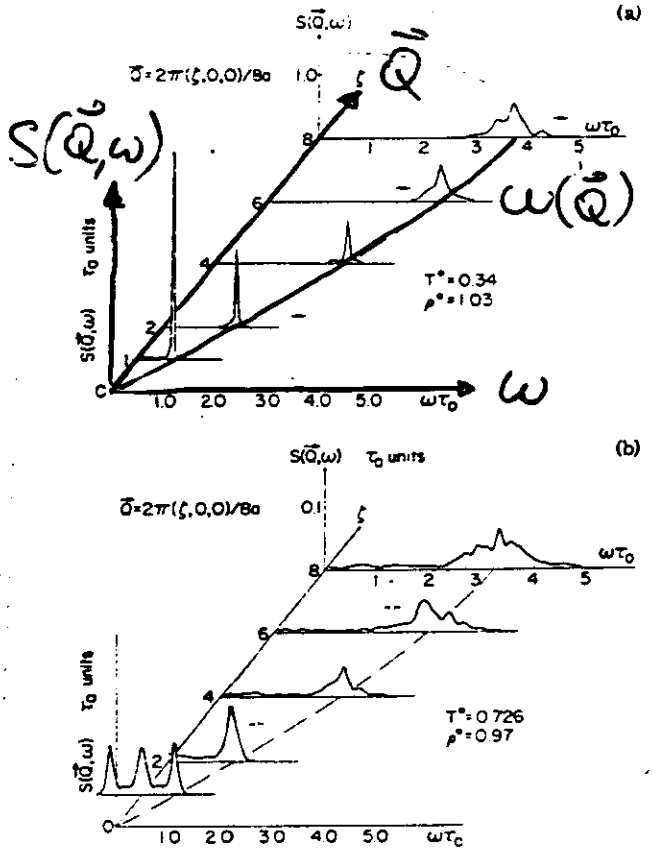


FIG. 4. (a) Dispersion curve for longitudinal phonons propagating in the [00ζ] direction taken from run No. 5. (b) Dispersion curve for longitudinal phonons propagation in the [00ζ] direction taken from run No. 10. Note the Rayleigh-Brillouin triplet for the smallest wave vector.

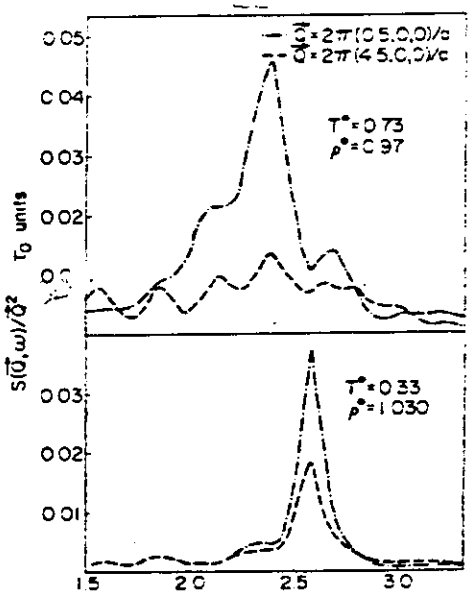


FIG. 5. $\vec{Q}$ dependence of $S(\vec{Q}, \omega)$ for the longitudinal phonon halfway to the zone boundary in the [00ζ] direction. The upper curve corresponds to roughly the triple point of the (12-6) potential while the lower curve corresponds to about one-half the melting point at zero pressure. Data taken from run Nos. 10 and 5, respectively.

F. Density shifts at constant temperature

Returning to Fig. 7 we see the strong negative shift in the longitudinal $(\frac{1}{3}, 0, 0)$ phonon peak position on changing the density from 1.031 to 0.97 at $T^* = 0.735$. The relative shift amounts to about 15% which compares well with predictions using

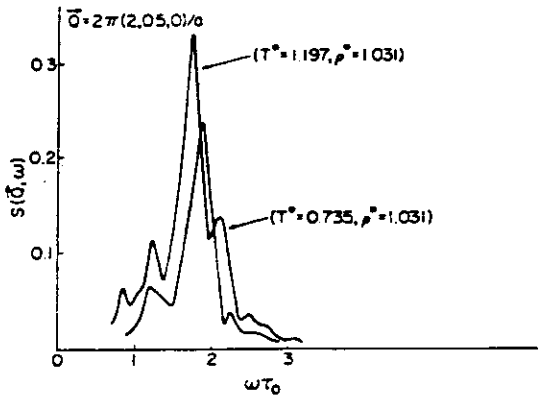


FIG. 6. Temperature dependence of $S(\vec{Q}, \omega)$ at constant density for the transverse phonon halfway to the zone boundary in the [00ζ] direction. Taken from run Nos. 13 and 14.

Dynamical mechanisms of atomic processes

MD : a window in the microscopic world

- Run ...
  look ...
  A-ha!
  improve theory.

Why

the Arrhenius plot of atomic self-diffusion in fcc metals is often

curved

near the melting point

?

C. Herzig et al.

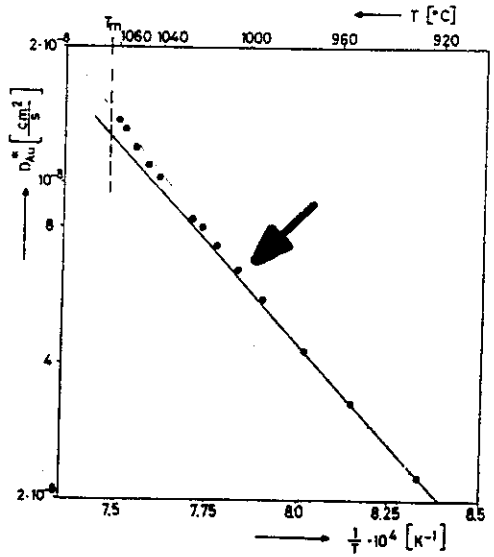


Fig. 1. Arrhenius plot of the high temperature diffusion coefficients of self-diffusion in gold single crystals. The solid line refers to the results in the temperature range between 758 and 993°C.

Migration of a vacancy in Au

G. DeLorenzi & F.E. (1992) [Europhys. Lett. 20, 343 (1992)]

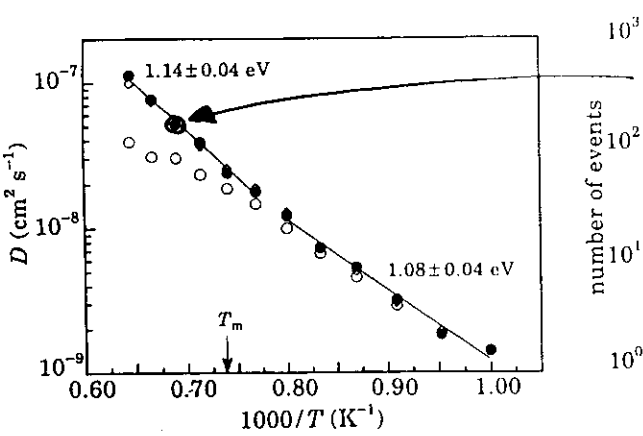


Fig. 1.

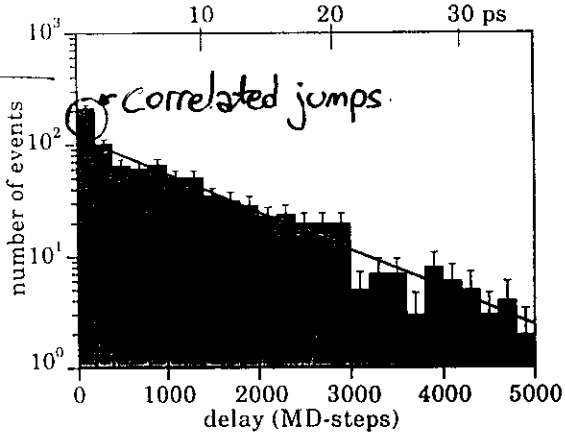


Fig. 2.

Fig. 1. – Atomic-migration contribution to the tracer diffusion coefficient D via monovacancies. Filled circles indicate D as measured directly from eq. (1) ($D = \langle u^2(t)\rangle - \langle u^2_{\text{bulk}}\rangle/6t$) diamonds indicate D_{all} as evaluated from the jump frequency Γ ($D_{\text{all}} = d^2 f_S \Gamma_{\text{all}}/6N$) with all the jumps treated as if they were single, uncorrelated jumps; and open circles indicate the contribution D_S from only single jumps ($D_S = d^2 f_S \Gamma_S/6N$). d is the nearest-neighbour distance and $f_S = 0.781\,46$ is the geometrical correlation factor for single jumps. Apparent activation energies are shown; $N = 256$.

Fig. 2. – Time delays between successive jumps of the vacancy. The bin width is 200 steps, $T = 1450$ K.

50

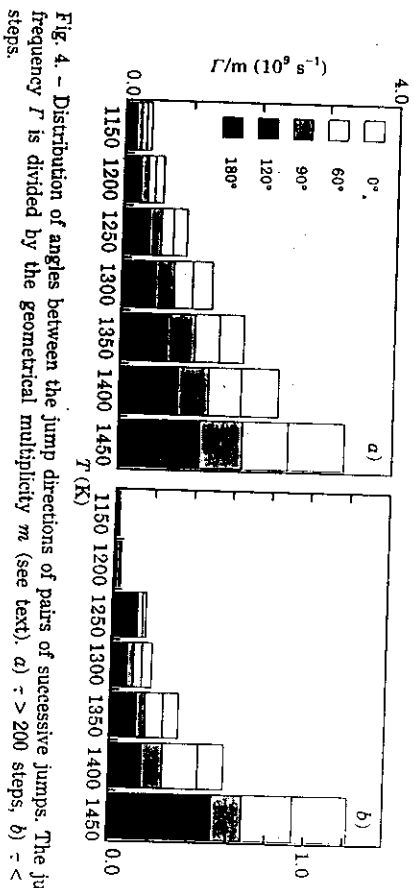


Fig. 4. – Distribution of angles between the jump directions of pairs of successive jumps. The jump frequency Γ is divided by the geometrical multiplicity m (see text). *a)* $\tau > 200$ steps, *b)* $\tau < 200$ steps.

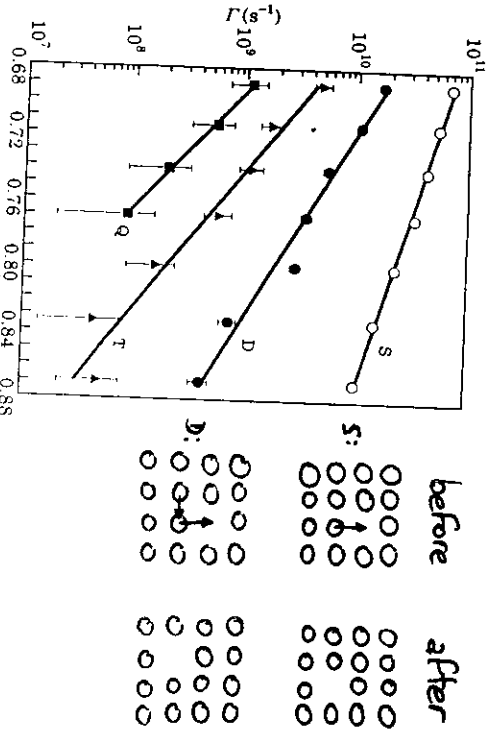
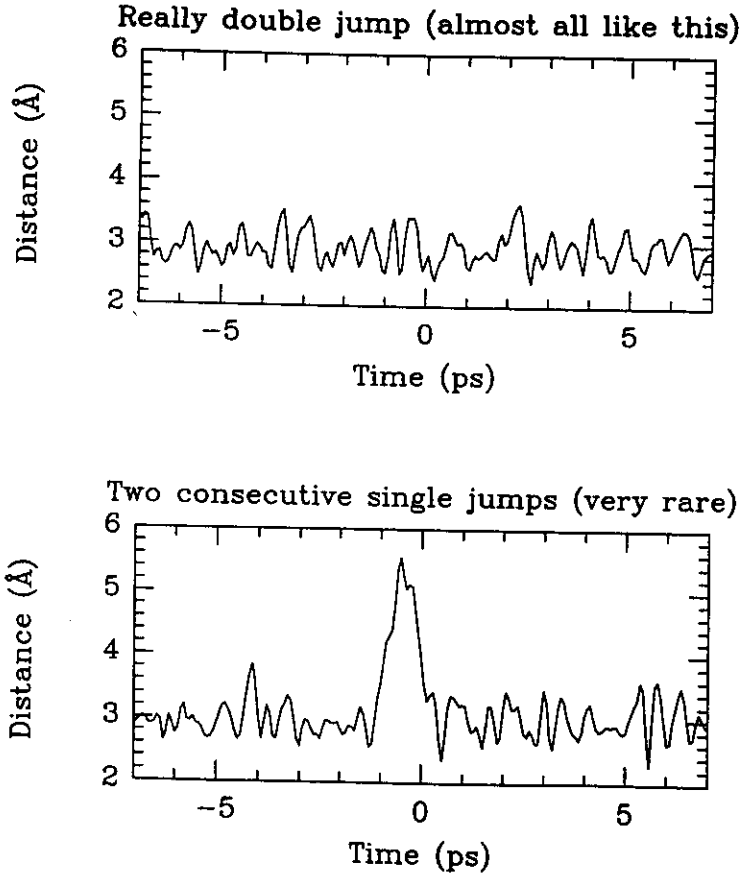


Fig. 3. – Frequencies of single (S), double (D), triple (T) and quadruple (Q) jumps: ■ (0.50 ± 0.02) eV, ● (1.64 ± 0.07) eV, ▲ (2.4 ± 0.2) eV, ■ (2.7 ± 0.8) eV. Error bars correspond to a standard deviation. The lines, whose slopes are given as activation energies, are obtained by a weighted fit.

50

(11)

Distance between the two jumping atoms in a 60° double jump [near Time=0]:



Migration of an atom adsorbed on a bcc(110) surface

G. De Lorenzi and G. Jacucci, 1985

with movie!

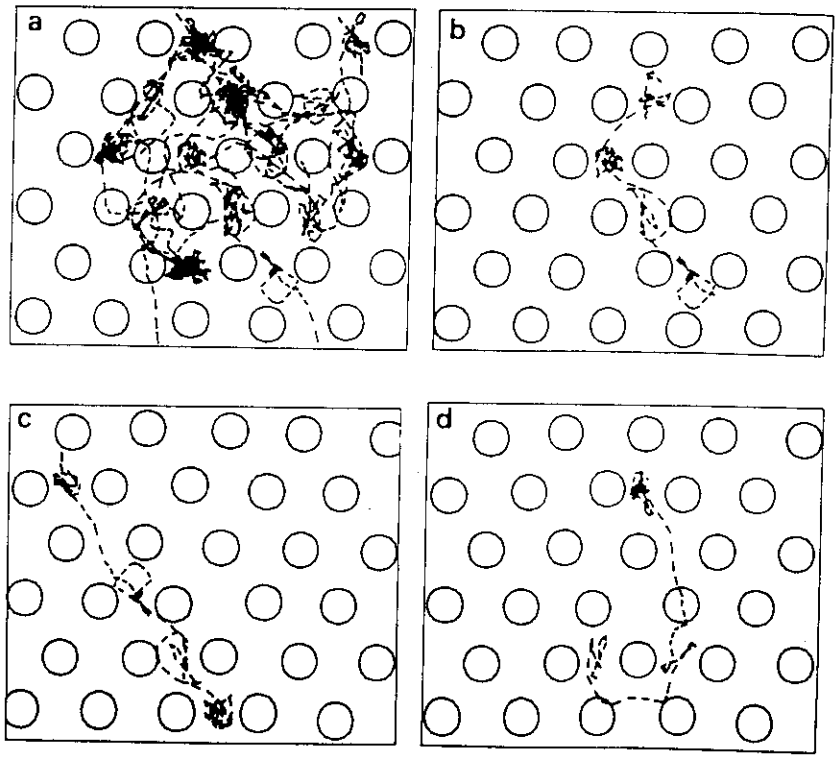
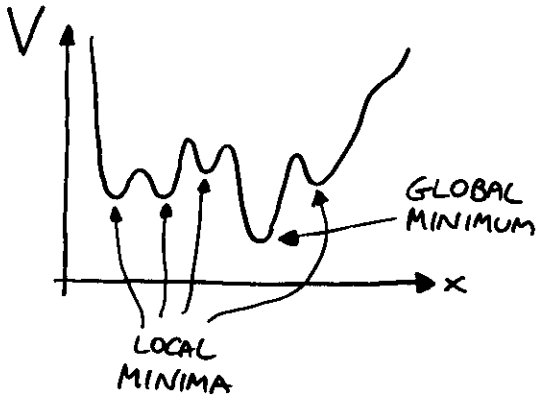
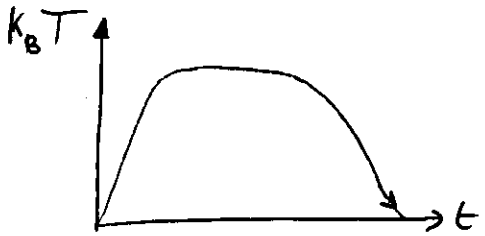


Fig. 4. Adatom migration on the bcc (110) surface, during a 200 ps interval (a). Short sketches from the same run are shown separately, in order to distinguish single jumps (b), double jumps (c) and more complicated trajectories (d). The positions of the surface atoms at the beginning of the interval are shown as open circles.

MD as an optimization tool



"Simulated annealing":



Work organization

- Plan run requirements (CPU, disk)
- Create subdirectories and document their contents
- Develop tools for handling chain of runs
  - less prone to errors once debug is made
  - less job babysitting ⇒ do more useful things
  - command files as documentation of sample history
  - rerun easily w/different size, different potential, etc.
  - insert automatic measurement devices ⇒ contraction of information
  - insert "intelligence" (may be dangerous)

What are potentials ?

$$H = \sum_i \frac{P_i^2}{2M_i} + \underbrace{\sum_n \frac{p_n^2}{2m} + \frac{1}{2}\sum_{ij}\frac{Z_i Z_j e^2}{|\vec{R}_i - \vec{R}_j|} + \frac{1}{2}\sum_{n'}\frac{e^2}{|\vec{r}_n - \vec{r}_{n'}|} - \sum_{in}\frac{Z_i e^2}{|\vec{R}_i - \vec{r}_n|}}_{H_{ee}}$$

$$\left[\sum_i \frac{P_i^2}{2M_i} + H_{ee}\right]\psi(\vec{R}_i, \vec{r}_n) = E\,\psi(\vec{R}_i, \vec{r}_n)$$

Born-Oppenheimer approximation (1923)

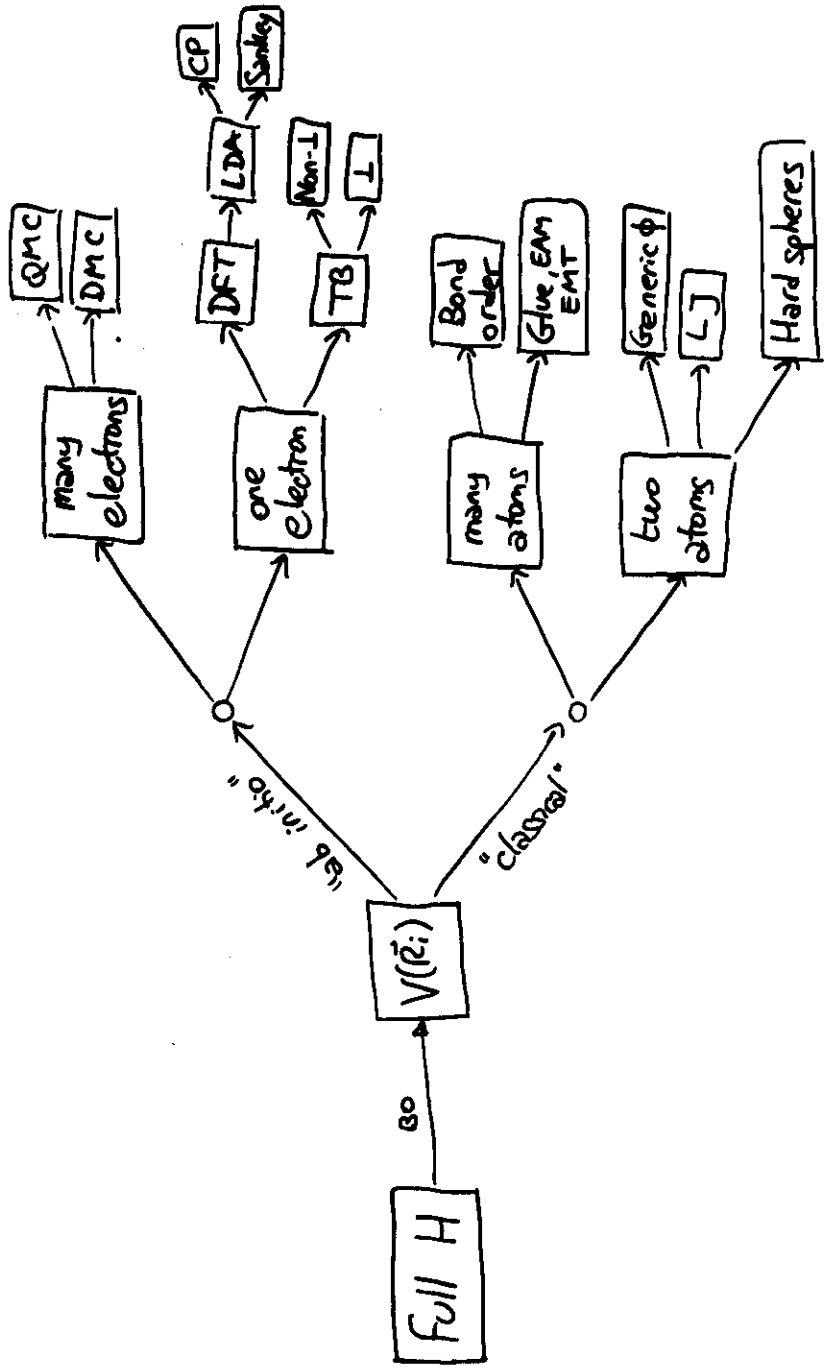
$m \ll M$ $\qquad \frac{\omega_e}{\omega_i} \sim \sqrt{\frac{M}{m}} \sim 100 \Rightarrow$ Regard nuclei as fixed for the electronic part

$$\psi(\vec{R}_i, \vec{r}_n) = \chi(\vec{R}_i)\,\phi(\vec{r}_n; \vec{R}_i)$$

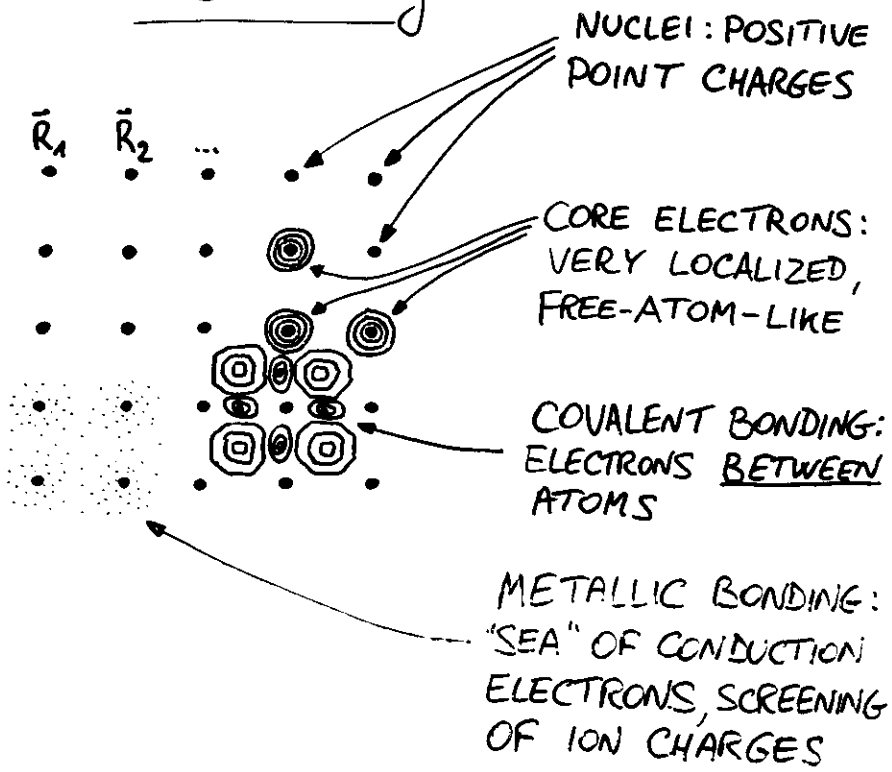
① $\quad H_{ee}\,\phi(\vec{r}_n; \vec{R}_i) = V(\vec{R}_i)\,\phi(\vec{r}_n; \vec{R}_i)$ $\qquad$ electrons

② $\quad \left[\sum_i \frac{P_i^2}{2M_i} + V(\vec{R}_i)\right]\chi(\vec{R}_i) = E\,\chi(\vec{R}_i)$ $\qquad$ nuclei

- $V(\vec{R}_i)$ is the potential function
- ② can be approximated classically
  [when $\Lambda = \sqrt{\frac{2\pi\hbar^2}{M k_B T}} \ll a$]



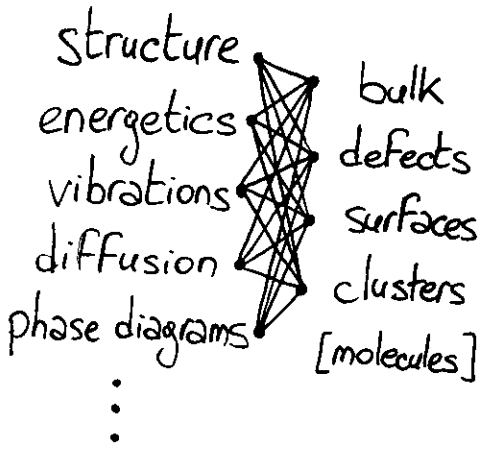
Bonding



$$V(\vec{R}_1, \ldots, \vec{R}_N) = ?$$

Potentials in the 90s:

realistic transferable fast



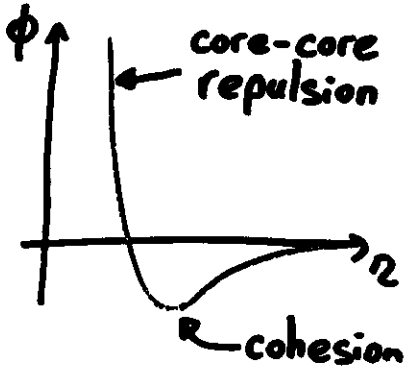
$N \sim 10^4 - 10^8$

$t \sim 1\text{ns} - 1\mu\text{s}$

- Right "chemistry"
- Multicomponent systems

Simplest scheme: pair potentials

$$V(\vec{r}_1,\dots,\vec{r}_N)=\frac{1}{2}\sum_{\substack{ij\\ i\neq j}}\phi(r_{ij})$$



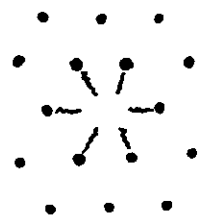
ϕ usually obtained by fitting
(cohesive energy, elastic constants, phonon frequencies, ...)

But in noble metals pair potentials do not work!

	Cu	Ag	Pt	Au	LJ
E_{coh}/k_BT_m	30	28	33	33	13
ΔS_m	1.2	1.1	1.2	1.1	~2
E_{vac}/E_{coh}	0.33	0.36	0.26	0.25	~1 ←
C_{12}/C_{44}	1.5	1.9	3.3	3.7	1

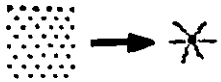
+ troubles on surfaces

Vacancy formation energy



Z atoms lose 1 bond

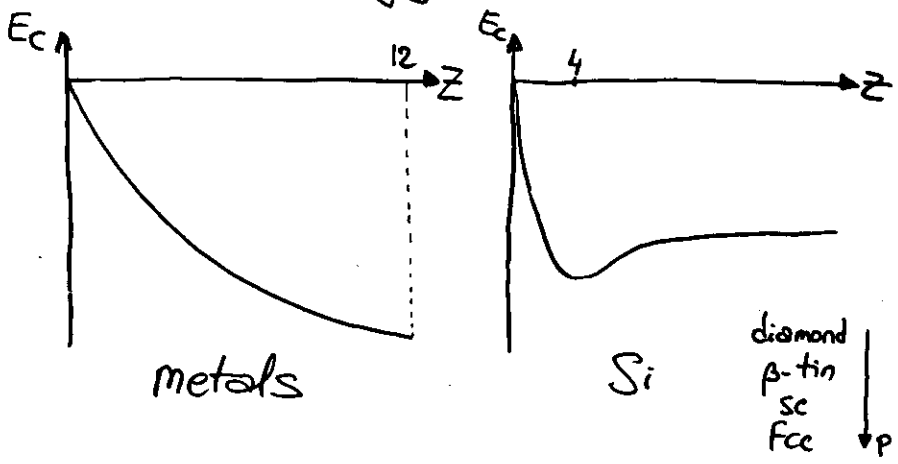
Cohesive energy



1 atom loses Z bonds

Two-body interactions ⇒ cost is the same

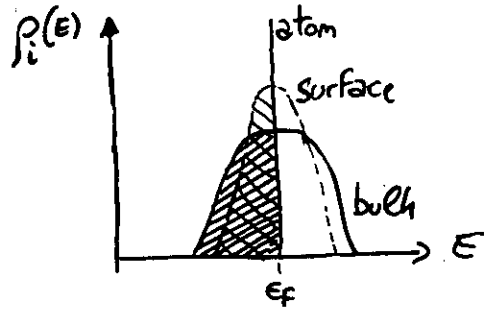
Cohesive energy vs coordination



Cohesion in strongly bonded systems does <u>not</u> have a two-body nature

Connection with tight-binding

[see A. Carlsson, Solid St. Phys. 43, 1 (1990)]



Bandwidth increases with coordination

$|i\rangle$: localized orbital

Assume $H = \sum_{ij} h_{ij}(r_{ij})\,|i\rangle\langle j|$

Forced to be <u>local</u>

Bandwidth approximately measured by $\sqrt{\mu_2^i}$

where μ_2^i is $\mu_2^i = \int_{-\infty}^{\infty} d\epsilon\, \epsilon^2 \rho_i(\epsilon)$

Apply algebra: $\rho_i(\epsilon) = \sum_n |\langle \psi_n | i\rangle|^2 \delta(\epsilon - \epsilon_n)$

where $H|\psi_n\rangle = \epsilon_n |\psi_n\rangle$

$$\mu_2^i = \int_{-\infty}^{\infty} d\epsilon\, \epsilon^2 \sum_n |\langle \psi_n | i\rangle|^2 \delta(\epsilon - \epsilon_n) =$$

$$= \langle i | \sum_n |\psi_n\rangle\langle\psi_n| \int_{-\infty}^{\infty} d\epsilon\, \epsilon^2 \delta(\epsilon - \epsilon_n) |i\rangle =$$

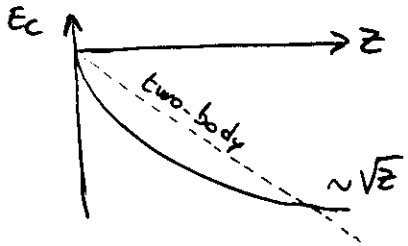
$$= \langle i | \sum_n \epsilon_n^2 |\psi_n\rangle\langle\psi_n | i\rangle = \langle i | H^2 | i\rangle =$$

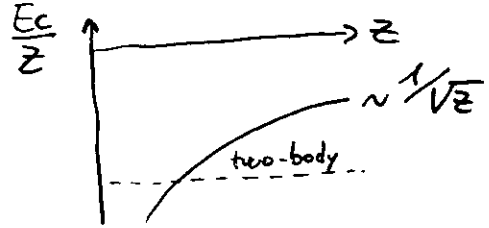
$$= \sum_j \langle i|H|j\rangle\langle j|H|i\rangle = \sum_j h_{ij}^2(r_{ij})$$

Result of connection with TB:

Cohesive energy $\propto$ bandwidth $\propto \sqrt{\mu_2^i} = \sqrt{\sum_j h_{ij}^2}$

$\propto \sqrt{Z}$





Vacancy formation energy:

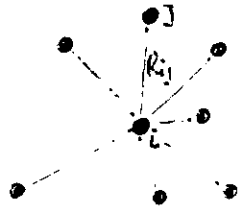
$$\frac{E_V}{E_C} \sim \frac{Z\cdot\left[\sqrt{Z}-\sqrt{Z-1}\right]}{1\sqrt{Z}} = Z\left(1-\sqrt{1-\frac{1}{Z}}\right)$$

$$\simeq Z\left(1-\left(1-\frac{1}{2Z}\right)\right) = \frac{1}{2}$$

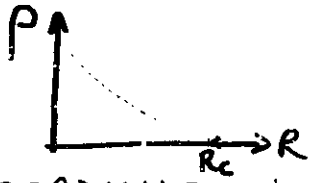
THE "GLUE MODEL"

$$V(\{\vec{R}_i\}) = \frac{1}{2}\sum_{\substack{ij\\(i\neq j)}} \phi(R_{ij}) + \sum_i U(n_i)$$

2-body — many-body ("glue")



$$n_i = \sum_{\substack{j\\(j\neq i)}} \rho(R_{ij})$$



n_i MEASURES COORDINATION

* MANY-BODY
* LOCAL NATURE $\Rightarrow$ SURFACES, DEFECTS
* DEPENDS ONLY FROM R_{ij} $\Rightarrow$ FAST MD SIMULATIONS 1000÷10000 ATOMS TIME ~ nsec

[M.W. Finnis and J.E. Sinclair (1984)
M.S. Daw, M.I. Baskes, S.M. Foiles (1984-86)
F. Ercolessi, M. Parrinello, E. Tosatti (1984-86)]

An invariance property

$$V = \frac{1}{2}\sum_{ij}\phi(r_{ij}) + \sum_i U(n_i)$$

$$n_i = \sum_j \rho(r_{ij})$$

Define

$$\phi^*(r) = \phi(r) + 2c\rho(r)$$
$$U^*(n) = U(n) - cn$$

Look at the glue potential given by ϕ^*, ρ, U^*:

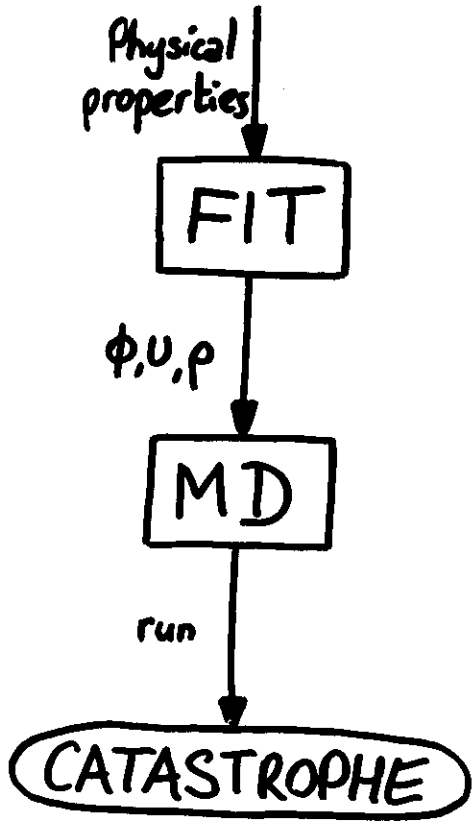
$$V^* = \frac{1}{2}\sum_{ij}\left[\phi(r_{ij}) + 2c\rho(r_{ij})\right] + \sum_i U(n_i) - c\sum_i n_i$$

$$= \frac{1}{2}\sum_{ij}\phi(r_{ij}) + c\sum_{ij}\rho(r_{ij}) + \sum_i U(n_i) - c\sum_{ij}\rho(r_{ij})$$

$$= V$$

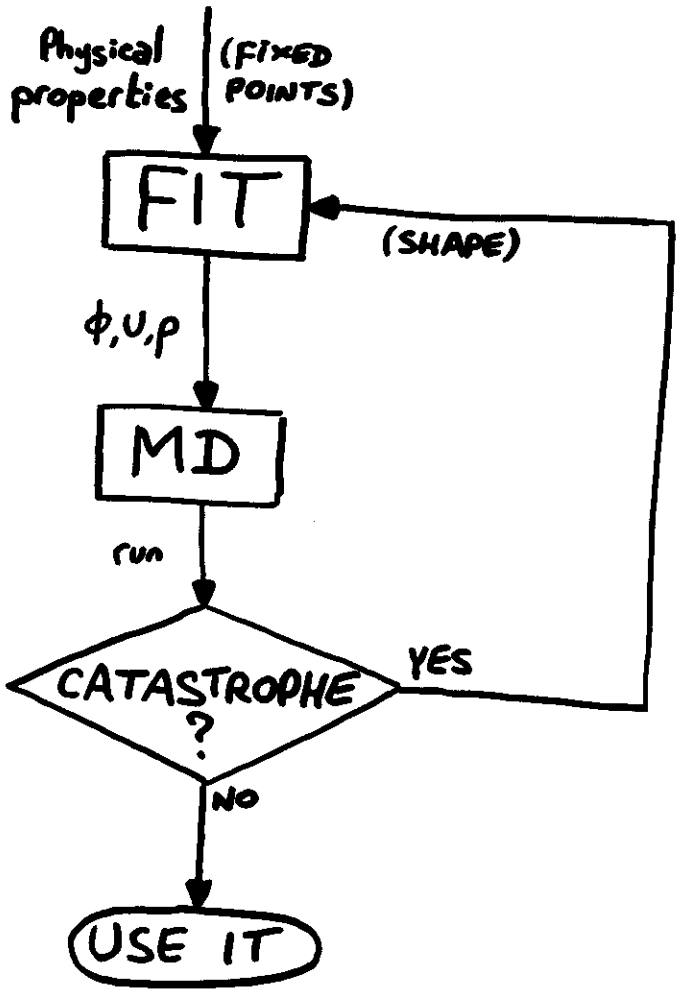
- Infinite # of equivalent potentials
- Physical properties only connected to invariants [such as $\psi(r,n) = \phi(r) + 2U'(n)\rho(r)$]

Usual result:



- Instability of crystal structures
- Wrong thermal properties (sometimes orders of magnitude!)
- Wrong surface properties (relaxations, reconstructions)
- Various unexpected things

A better scheme



The glue model

- Classical Hamiltonian based on the following expression:

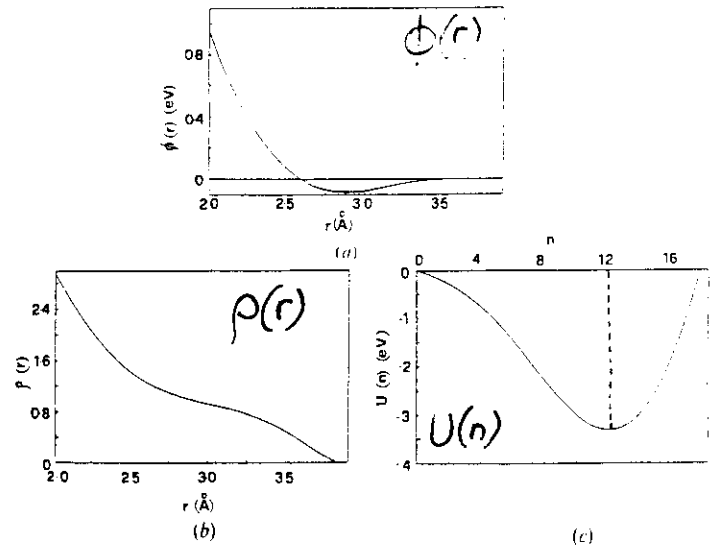
$$V = \frac{1}{2}\sum_{ij}\phi(r_{ij}) + \sum_{i}U(n_i)$$

where n_i is a "coordination number"

$$n_i = \sum_{j}\rho(r_{ij})$$

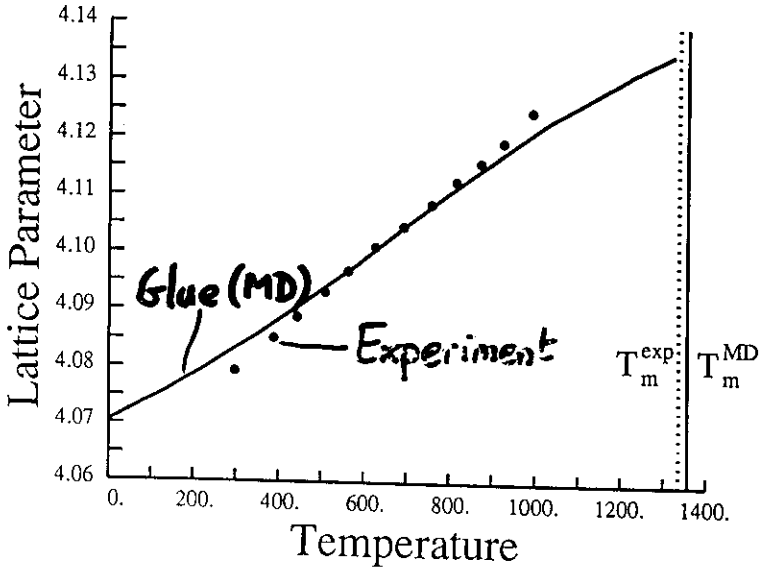
- Parametrization for Au:

(Phil. Mag. A 58, 213 (1988))



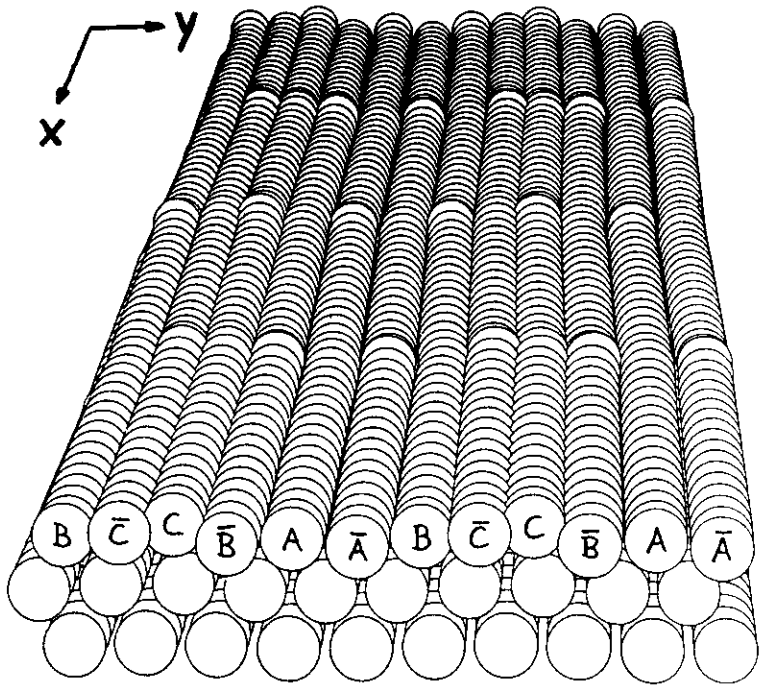
- Reproduces surface reconstructions of Au(100), Au(110), Au(111)

Thermal expansion and melting temperature



Temperature-dependent properties are fitted by acting on the shape of the functions

Au(100) reconstruction



THEORY : 34×5
EXPT : ~28×5

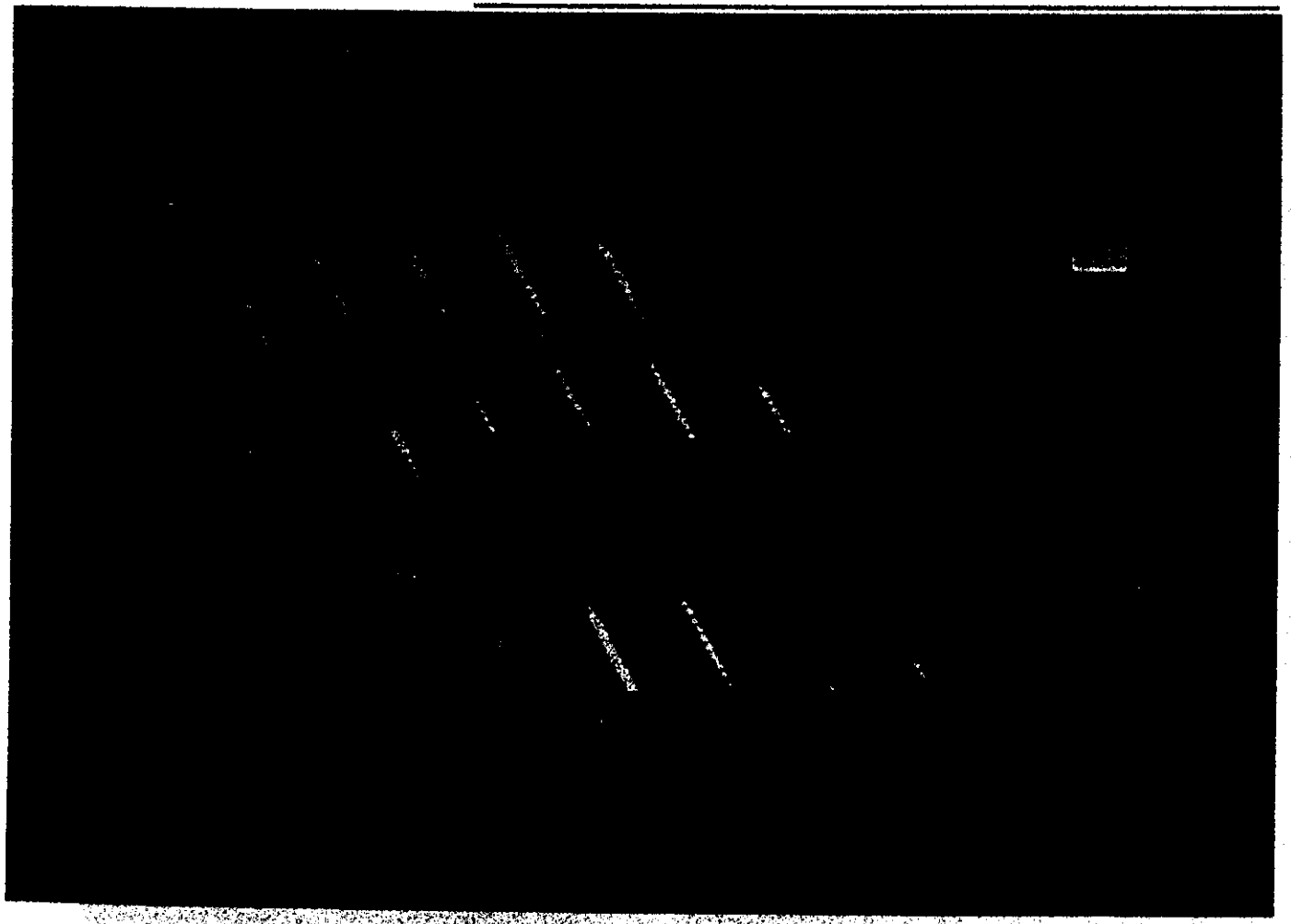
Phys Rev. Lett. 57, 719 (1986)

F9

R39

Fig 4

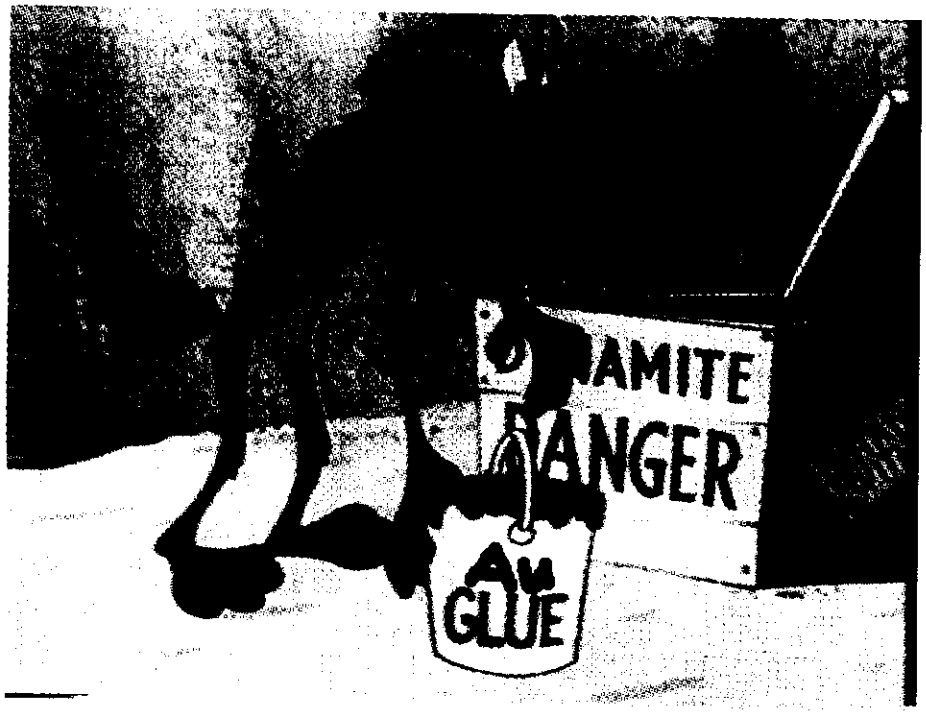
Picture 1



3-D Plot of a Au (100) face

gives an impressive insight to the complex structure of the reconstructed surface. Chains of gold atoms with a long range modulation of height are the dominating feature.

Other applications:



Wile E. Coyote, Dept. of Ornithology, Arizona State Univ.

(work in progress)

How to compute glue forces

$$\vec{F}_i = -\sum_{j\neq i}\left\{\frac{\phi'(r_{ij})}{r_{ij}} + \left[U'(n_i)+U'(n_j)\right]\frac{\rho'(r_{ij})}{r_{ij}}\right\}\vec{r}_{ij}$$

($n_i = \sum_k \rho(r_{ik})$, $n_j = \sum_k \rho(r_{jk})$)

① compute n_i's

```
do i=1,N
  do j∈{neighbors of i}
    compute r_ij^2
    get ρ(r_ij) from tables
    n_i = n_i + ρ(r_ij)
    n_j = n_j + ρ(r_ij)
  enddo
enddo
```

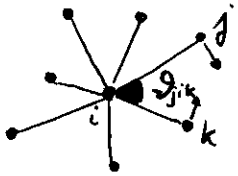
② get $U'(n_i)$ from n_i

```
do i=1,N
  get U'(n_i) from tables
enddo
```

③ compute $\vec{F}_i$

```
do i=1,N
  do j∈{neighbors of i}
    compute r_ij^2
    get φ'(r_ij)/r_ij from tables
    get ρ'(r_ij)/r_ij from tables
    F_i = F_i - {φ'(r_ij)/r_ij + [U'(n_i)+U'(n_j)] ρ'(r_ij)/r_ij} r_ij
    F_j = F_j + {          same          } r_ij
  enddo
enddo
```

Glue potentials: no true angular forces



$$V=\frac{1}{2}\sum_{ij}\phi(r_{ij})+\sum_i U\left(\sum_j \rho(r_{ij})\right)$$

Covalent systems => problems

Covalent systems: geometry-driven vs electronic structure-driven Fits

Stillinger-Weber For Si (1985):

$$V=\frac{1}{2}\sum_{ij}\phi_2(r_{ij})+\sum_{ijk} g(r_{ij})g(r_{ik})\left(\cos\vartheta_{jik}+\frac{1}{3}\right)^2$$

Force Fields (for biomolecules):

$$V=\frac{1}{2}\sum_{ij}\phi_2(r_{ij})+$$  + ...

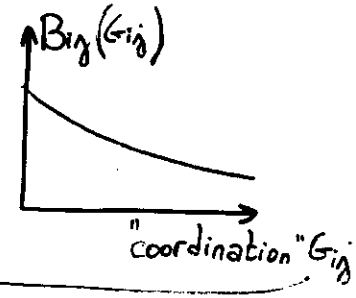
$K_{\vartheta\vartheta'\phi}(\vartheta-\vartheta_0)(\vartheta'-\vartheta'_0)\cos\phi$

Semiconductors: "bond order" potentials

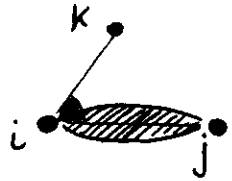
$$V(\vec{R}_1,..\vec{R}_N)=\frac{1}{2}\sum_{ij}\phi_R(R_{ij})+\frac{1}{2}\sum_{ij}B_{ij}\,\phi_A(R_{ij})$$

B_{ij} depends upon environment

Abell, Tersoff (1985-88)



$$G_{ij}=\sum_{-k} F_c(R_{ik})\,g(\vartheta_{jik})\,F(R_{ij}-R_{ik})$$



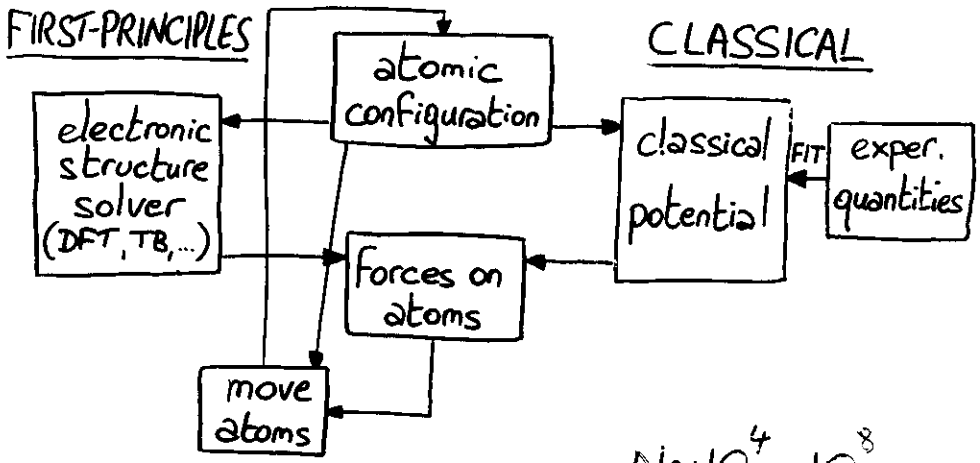
Other bonds ik weaken ij

- Can be connected with first-principle via tight-binding
- Work fairly well for Si, C

Problems

- Difficult to get phonons and surfaces right
- Coordination of liquid Si is too low
- Fit is complicate. Done with ~12 parameters

Molecular dynamics



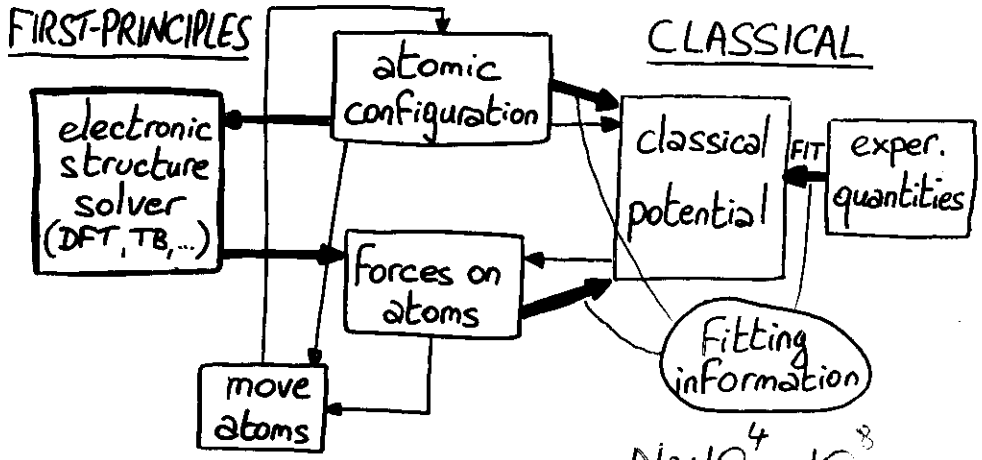
Problem: heavy
$N \sim 100-1000$
$t \sim$ ps

$N \sim 10^4 - 10^8$
$t \sim$ ns

Problems:

- Transferability (surfaces, clusters, molecules, ...)
- Finite T (th. expansion, melting, liquid, ...)
- Realistic models => complex analytic forms => Fitting difficult and unpleasant

Molecular dynamics



Problem: heavy
$N \sim 100-1000$
$t \sim$ ps

$N \sim 10^4 - 10^8$
$t \sim$ ns

The force-matching method tries to eliminate them

Problems:

- Transferability (surfaces, clusters, molecules, ...)
- Finite T (th. expansion, melting, liquid, ...)
- Realistic models => complex analytic Forms => Fitting difficult and unpleasant

Force matching method: F.E. and J.B. Adams, Europhys. Lett. 26, 583 (1994)

Goals:

* **Use *large* set of ab initio data to obtain good transferability**

* **Rich analytical forms *and* powerful parametrization (full shape control)**

* **Correct first principles problems**

* **Still use expt data where needed**

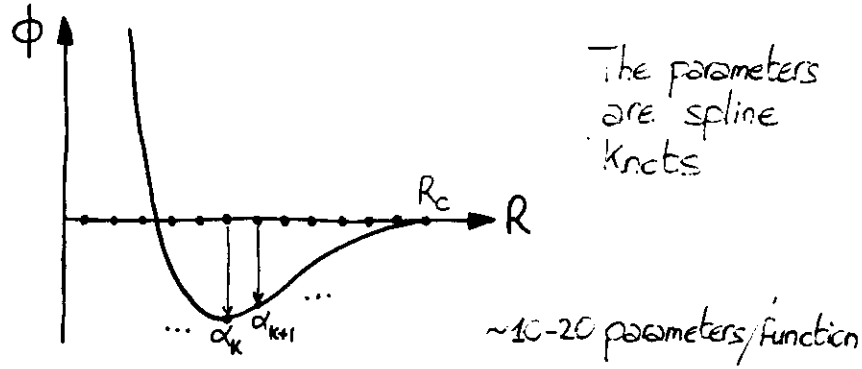
Parametrization: powerful

Potentials are usually based on one-variable functions

Example: EAM is made of three functions

$$V(\{\mathbf{R}\};\alpha) = \frac{1}{2}\sum_{ij}\phi(R_{ij};\alpha) + \sum_{i}F(n_i;\alpha)$$

with $\quad n_i = \sum_j \rho(R_{ij};\alpha)$



Optimization: by minimization

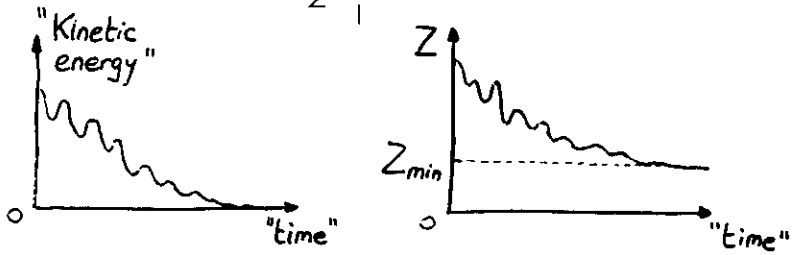
$$Z(\alpha) = (\text{const}) \sum_k \sum_i |\mathbf{F}_{ki}(\alpha) - \mathbf{F}^{\circ}_{ki}|^2$$

$$+ \sum_r W_r [A_r(\alpha) - A^{\circ}_r]^2$$

- Force matching
- Other physical constraints

Simulated annealing:

$$\mathcal{L} = \frac{1}{2} \sum_l m_l \dot{\alpha}_l^2 - Z(\alpha)$$



Why aluminum?

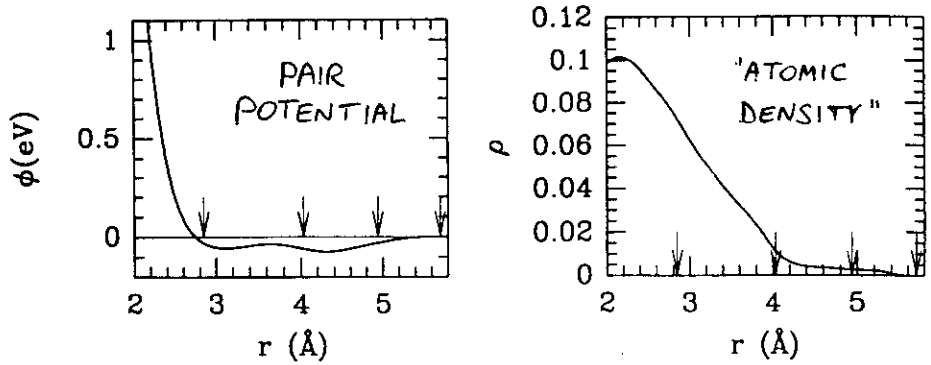
* It is a benchmark for metals
* Easy to do by first-principles (no d electrons)
* Use pair functional potential (EAM-like): simple
* Existing potentials have problems:

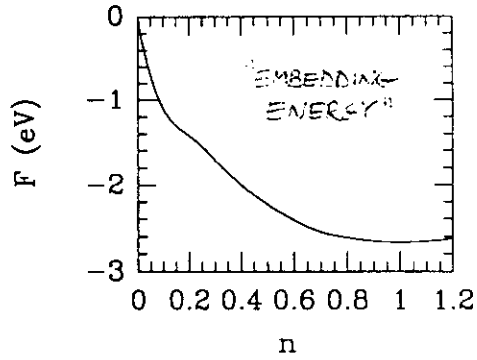
E_v^m E_{SF} γ T_m L α

Details

- Sankey's DFT-LDA real space MD code
- 85 configurations as input (bulk with vacancy, liquid, (100), Al_{150}) [about 10000 force vectors in Z]
- Output: 3rd neighbour glue, 40 parameters
- 32 external constraints [E_{coh}, a, C_{11}, C_{12}, C_{44}, E^f_{vac}, E_{SF}, γ_{111}, and Rose-Smith-Guinea-Ferrante equation]
- Simulated annealing once at the beginning. Direct minimization for adjustments.
- Some trial and error using MD to check thermal properties

Final potential



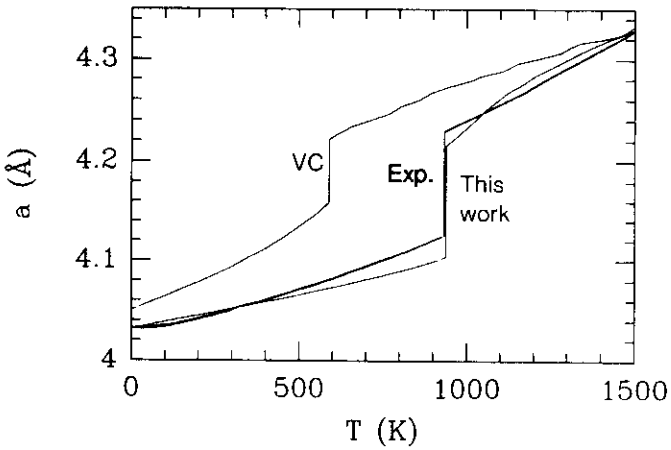


$$\left[\begin{array}{c} n_0 = \sum_{\vec{R}} \rho(R) \equiv 1 \\ F'(n=1) = 0 \end{array}\right]$$

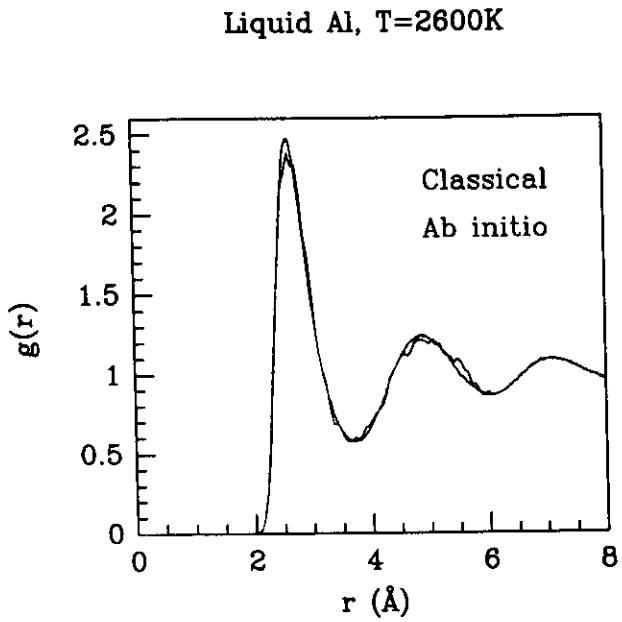
$\frac{\sqrt{\langle (F-F^0)^2 \rangle}}{\sqrt{\langle F^{0\,2} \rangle}}$:	VC	FD
18%	36%	45%

	Exp	This	VC	FD
E_v^m(eV)	.62	.61	.30	.54
E_{SF}(meV/A^2)	~8	6.5	4.7	2.9
γ_{100}(meV/A^2)	71	59	53	34
T_m (K)	933	939	590	740
L (meV/atom)	108	105	53	98

Thermal expansion and melting



[$a \equiv (4\Omega)^{1/3}$ in the liquid]



Conclusions

The force-matching method : a tool to achieve

- accuracy comparable with ab initio methods (by "learning" from them)
- high transferability (by sampling)

Function shapes determined automatically

$\Rightarrow$
- easy to use rich analytical expressions
- compare them quantitatively

Future : more materials
other Functional forms
tight binding