

**Computational Physics**  
**Modern Techniques in Materials Science**

V. Chirita

Thin Film Physics Division  
University of Linköping

## **Course Outline**

- **Scientific Foundation of Computer Simulations**
- **General Features & Common Techniques**
- **Molecular Dynamics (MD) Simulations**
- **Monte Carlo (MC) Simulations**
- **Applications in Materials Science: Thin Films**

# WHY do we need computer simulations ?

- Today, state-of-the-art experimental techniques provide resolutions down to the angstrom scale:
  - Electron microscopy (HRTEM, SEM)
  - Scanning Tunnelling Microscopy (STM)
  - Field Ion Microscopy (FIM)
  - Atomic Force Microscopy (AFM)
  - Diffraction techniques (XRD, LEED, RHEED)
- Theoretical models give us a solid understanding of the governing laws in nature and its building blocks:
  - quarks, nucleus, electrons, atoms, elements, galaxies, universe.

Experiments provide detailed information about **initial** and **final** states, but basically **no information** about **transient** processes.

- How are atoms, molecules, or clusters, moving ?

Theory gives us usually equations, describing simple systems. The exact analytical solutions are only known for 2-body problems. **There is no exact solution for many-bodies problems.**

In materials science, problems involve **many-bodies**, often in **transient states**.

- N particles ( $N \gg 2$ )
- typical time scales: fs – ps – ns range.

**Computer simulations are essential tools** in materials science.

- **overcome** experimental difficulties (time/space resolution).
- **solid** test for theoretical models.
- **bridge** between theory and experiment.
- **yield** realistic predictions (materials science, ecology, biology, economics).

## Computer simulations:

- **do not** replace experiments.
- **are not** as general as pure theoretical treatments.
- **complement** experiments and theoretical models.

## Advantages:

- **implementation**
- *experimental inputs*
- **reproducibility**
- **visualisation**
- relatively easy & inexpensive.
- easily changeable & varied.
- practically 100 % reproducible.
- offer unique information at atomic level.

## Limitations:

Two main restrictions, based on CPU speed & memory:

- **size** of systems simulated.
- **time scale** of simulations.

## WHAT is a simulation ?

A simulation has some *state* variables –  $\mathbf{S}$ . In classical mechanics,  $\mathbf{S}: (q_i, p_i)$ .

Initial state  $\mathbf{S}_0 \rightarrow$  modified by some *process*  $\rightarrow$  new state  $\mathbf{S}_{n+1} = T(\mathbf{S}_n)$

$T(\mathbf{S})$  = iteration function (deterministic or stochastic).

$n$  = iteration index representing *time* (fictitious time).

Goal of simulations is to obtain a set of properties in equilibrium:  $\mathbf{A}(\mathbf{S})$ .

-typical examples is internal energy  $E(\mathbf{S})$ ,  $T(\mathbf{S})$ ,  $P(\mathbf{S})$  etc.

Compare with properties measured in real experiments:  $\langle \mathbf{T} \rangle$ ,  $\langle \mathbf{P} \rangle$  etc.

$\langle \rangle$  averages over large **number** of particles and **time** of measurement.

### QUESTION:

how do we connect microscopic  $(q_i, p_i)$  with macroscopic  $(\langle \mathbf{A} \rangle)$  properties ?

The physical foundation of simulation is **statistical mechanics**.

Real experiments yield  $\langle \mathbf{A} \rangle$  as a time average, from measurements made at successive times ( $t_1, t_2, \dots, t_M$ ):

$$\langle \mathbf{A} \rangle = \sum_i \mathbf{A}_i \mathbf{P}_i \text{ where } \mathbf{P}_i = \mathbf{n}_i / \mathbf{M} \text{ is number of times system is in state } i.$$

Gibbs introduced the artifice of large number of replicas of a system (*ensemble*) and replaced  $\langle \mathbf{A}_{\text{time}} \rangle$  with  $\langle \mathbf{A}_{\text{ens}} \rangle$  :

$$\langle \mathbf{A}_{\text{time}} \rangle = \langle \mathbf{A}_{\text{ens}} \rangle$$

The equality holds only if the system is **ERGODIC**.

**Ergodicity** is the *fundamental* assumption of classical statistical mechanics.

The **ergodic hypothesis** is also the basis of **computer simulations**.

## WHAT is Ergodicity?

- \* after a certain time, one loses memory of ones initial state.
- \* systems, no matter how prepared, relax after reasonable time towards statistical equilibrium (all macroscopic variables are constant in time).
- \* time average over a typical trajectory equals the ensemble average.

**Attention!** Not all systems are **ERGODIC!!!**

Non-ergodic behaviour = equilibrium is reached depending of initial conditions.

- typically a *low temperature* effect.
- characteristic for *classical systems*.
- example: 64 Ar atoms at 5K will never reach equilibrium!

In computer simulations, non-ergodicity allows the study of metastable states occurring in nature (for example, freezing of super-cooled liquids or glass).

Any computer simulation starts with a **valid statistical ensemble**.

**valid = each thermodynamic variable is fixed**

microcanonical	-----	constant <b>N, V, E</b>
canonical	-----	constant <b>N, V, T</b>
isothermal-isobaric	-----	constant <b>N, P, T</b>
grand canonical	-----	constant <b><math>\mu</math>, V, T</b>

Often, one should include some contact with a thermal bath.

All simulation algorithms can sample each ensemble distribution.

<b>Molecular dynamics (MD)</b>	- no randomness ( <i>deterministic</i> )	$\rightarrow \langle A \rangle_{\text{time}}$
<b>Langevin Dynamics</b>	- heat bath adds additional forces.	
<b>Brownian Dynamics</b>	- heat bath determines velocities.	
<b>Smart Monte Carlo</b>	- random walk biased by force.	
<b>Monte Carlo (MC)</b>	- unbiased random walk ( <i>stochastic</i> )	$\rightarrow \langle A \rangle_{\text{ens}}$

**MD approach  $\Rightarrow \langle A \rangle_{\text{time}} \sim \langle A \rangle_{\text{ens}} \Leftarrow$  MC approach**

## General features & common techniques

We concentrate on Monte Carlo (MC) and Molecular Dynamics (MD) methods.

Classical system with  $N$  particles  $\rightarrow (q_i, p_i), i = 1, 2, \dots, N \equiv$  classical phase space.

Within the B.O. in the classical limit:  $(q_i, p_i) \Leftrightarrow H(q_i, p_i) = K(p_i) + V(q_i)$

**H – the hamiltonian is the basic input in any simulation.**

$q_i \equiv r_i$  – cartesian coordinates;  $p_i$  – conjugate momenta

**Kinetic Energy:** 
$$K = \sum_{i=1}^N \sum_{\alpha} p_{i\alpha}^2 / 2m_i; \quad \alpha = x, y, z$$

**Potential Energy:** 
$$V = \sum_i v_1(r_i) + \sum_i \sum_{j>i} v_2(r_i, r_j) + \sum_i \sum_{j>i} \sum_{k>j>i} v_3(r_i, r_j, r_k) + \dots$$
  
$$\begin{array}{ccc} \parallel & \parallel & \parallel \\ \text{Ext. field} & \text{Pair potential} & \text{3-body term} \end{array}$$

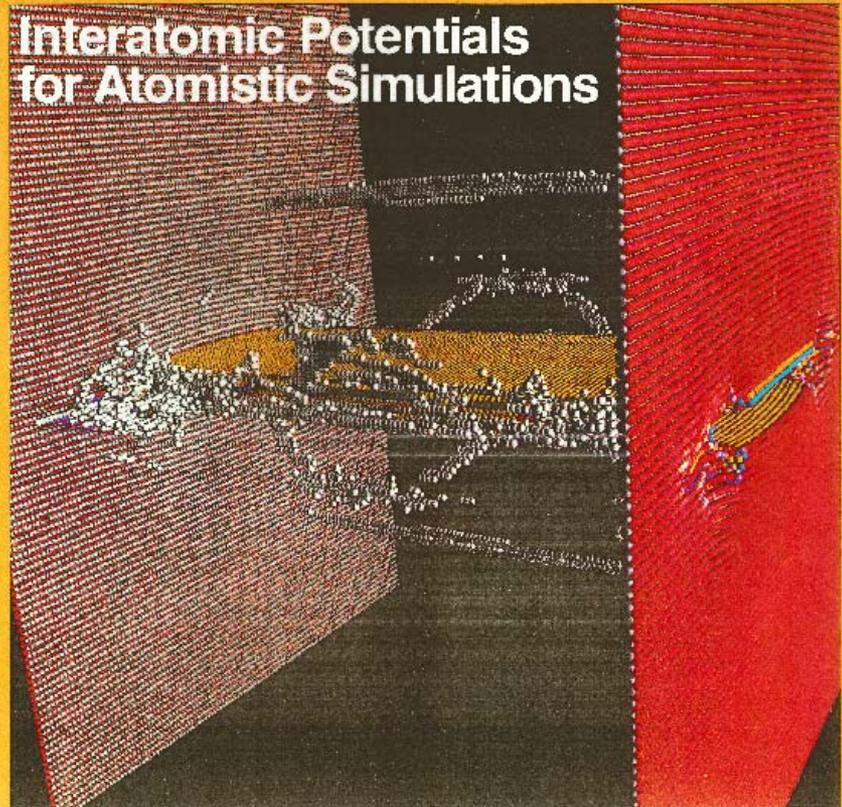
**MRS**  
**BULLETIN**

Serving the International  
Materials Research Community  
A Publication of the Materials Research Society

February 1995, Volume 21, No. 2



**Interatomic Potentials  
for Atomistic Simulations**



Exceptionally active and  
important research field

# Interaction Potentials

Essential feature, requirement for MD and MC correct computations.

Results, accuracy of predictions depend massively of interaction quality, exclusively in the case of MD simulations.

Consequently, **interaction potentials** are **the most important ingredient** in classical MD and MC simulations.

Most general formulation for potential energy of a classical system depending on the coordinates of the atoms:

$$E_{\text{total}} = \sum_{i<j} \Phi_2(\mathbf{r}_i, \mathbf{r}_j) + \sum_{i<j<k} \Phi_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \dots + \sum_{i<j<k<\dots<n} \Phi_n(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k, \dots, \mathbf{r}_n) + \dots$$

where  $n = 2, 3, 4, \dots$  and  $\sum_{i<j<k<\dots<n} \Phi_n(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k, \dots, \mathbf{r}_n)$  is the sum of the ***n*-body** terms.

Fundamental assumption is that the series converges rapidly & higher order terms are neglected. Based on this expansion, interaction potentials can be classed as:

- **Pair potentials**, exclusively two-body terms -  $\Phi_2(\mathbf{r}_i, \mathbf{r}_j)$ .
- **Many-body potentials**, three-body and higher terms.

In practice, the *n*-body expanded is usually **truncated** after  $\Phi_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k)$ .

## Criteria for Choosing a Potential

- \* **Accuracy** - reproduce properties as closely as possible.
- \* **Transferability** - applicable to as many systems as possible.
- \* **Computational speed** - force calculations are the most time consuming.
  - 2-body potential  $V(r_{ij}) = V(\|r_i - r_j\|)$
  - 3-body potential  $V(r_{ijk})$  depends on orientation – more computation.

## Typical Emphasis for Various Disciplines

1. **CHEMISTRY** – priority is on accuracy.
  - rate constants, for example, require very accurate reaction barriers.
2. **STATISTICAL MECHANICS** – Computational speed is important.
  - Complexity can easily arise even simple potentials are used.
3. **BIOCHEMISTRY** – Combination of all criteria is desired.
  - Protein structures, polymers have significantly different configurations.
4. **MATERIALS SCIENCE** – most certainly all three criteria.
  - Calculation of equilibrium atom positions, comparison with experiment.

## Obtaining a Potential

1. Analytic potentials based on ad hoc functional forms and assumptions.
  - typically, pair potentials
2. Analytic potentials derived from quantum-mechanical bonding arguments.
  - Embedded Atom Method (EAM)
  - Bond order potentials
  - Effective Medium Theory (EMT)
  - Finnis and Sinclair (FS)
  - Glu model
  - **all** appear as 2-body-type, in reality, cleverly disguised MANY-body potentials.
3. Forces obtained directly from electronic structure (*ab-initio*) calculations.
  - Car-Parrinello MD
4. Force Field Methods – extremely valuable but never transferable.
  - Can contain accurate parameterizations for intramolecular, as well as pair (LJ) and long range (Coulomb) interactions, but only for one material.

# Potential Functionals

\* 2-body interactions:  $\Phi(1, \dots, N) = \sum_{i < j} V_2(i, j)$

- work fine for rare gases (Ar, Ne, Kr, Xe).

- **Lennard-Jones**:  $V_2(r) = 4\varepsilon [x^{-12} - x^{-6}]$

$\varepsilon$  - maximal well-depth of potential.

$x = r/\sigma$  - distance in units of  $\sigma$ , where  $\sigma$  is effective radius.

- **Morse, Yukawa, Double Yukawa, Hard-Sphere, Soft-Sphere** etc.

\* 3-body interactions:  $\Phi(1, \dots, N) = \sum_{i < j} V_2(i, j) + \sum_{i < j < k} V_3(i, j, k)$

- used mostly for covalent systems (Si, C, SiC).

- **Stillinger – Weber**:

$V_2(r) = A (Br^p - r^q) \exp[(r-a)-1]$ ; where A, B, p, q, a – params.

$V_3(r) = h(r_{ij}, r_{ik}, \theta_{jik}) + h(r_{ji}, r_{jk}, \theta_{ijk}) + h(r_{ki}, r_{kj}, \theta_{ikj})$ .

- **Tersoff**:

$V_2(r) = f_C(r) + [f_R(r) + b f_A(r)]$ ;  $b = b(\theta_{ijk})$

## Tersoff functional

- the 2 & 3 body interactions require extensive CPU time.

$$E = \sum_i^N \Phi_i; \quad \Phi_i = \frac{1}{2} \sum_{j(\neq i)}^N f_c(r_{ij}) [V_R(r_{ij}) - b_{ij} V_A(r_{ij})]$$

where

$$f_c(r_{ij}) = \begin{cases} 1 & \text{if } r_{ij} < R - D \\ \frac{1}{2} - \frac{1}{2} \sin \left[ \frac{\pi}{2} (r_{ij} - D) / D \right] & \text{if } R - D < r_{ij} < R + D \\ 0 & \text{if } r_{ij} > R + D \end{cases}$$

3-body part given by:

$$V_R(r_{ij}) = A \exp(-\lambda_1 r_{ij}); \quad V_A(r_{ij}) = B \exp(-\lambda_2 r_{ij}); \quad b_{ij} = (1 + \beta^n \xi_{ij}^n)^{-1/2n}$$

$$\xi_{ij} = \sum_{k(\neq i, j)}^N f_c(r_{ik}) g(\theta_{ijk}) \exp[\lambda_3^3 (r_{ij} - r_{ik})^3]$$

$$g(\theta_{ijk}) = 1 + c^2 / d^2 - c^2 / [d^2 + (h - \cos \theta_{ijk})^2]$$

A, B,  $\lambda_1$ ,  $\lambda_2$ ,  $\lambda_3$ ,  $\beta$ , n, c, d, h – given parameters; R, D [Å] – interaction range

## Embedded-Atom Method (EAM) – Johnson's functional

- very useful for metal-metal interactions (this is the fcc analytical form);
- there are also separate forms for bcc and hcp structures.

$$E_t = \sum_i F(\rho_i) + \frac{1}{2} \sum_{i,j}' \Phi(r_{ij}); \quad \rho_i = \sum_j' f(r_{ij})$$

- where:
- $E_t$  = total internal energy.
  - $\rho_i$  = the electron density at atom  $i$  due to all other atoms.
  - $f(r_{ij})$  = the electron density at atom  $i$  due to atom  $j$  as a fct. of distance.
  - $r_{ij}$  = separation distance between atoms  $i$  and  $j$ .
  - $F(\rho_i)$  = the energy to embed atom  $i$  in an electron density  $\rho_i$ .
  - $\Phi(r_{ij})$  = two-body potential between atoms  $i$  and  $j$ .

- can also be used in the following form for all alloy potentials:

$$\Phi^{ab}(r) = \frac{1}{2} \left( \frac{f^b(r)}{f^a(r)} \Phi^{aa}(r) + \frac{f^a(r)}{f^b(r)} \Phi^{bb}(r) \right)$$

- electron density and two-body potential taken as decreasing exponentials:

$$f(r) = f_e \exp[-\beta(r/r_e - 1)]; \quad \Phi(r) = \Phi_e \exp[-\gamma(r/r_e - 1)].$$

-the embedding function is determined by fitting to a universal eq. of state in the form of a Rydberg fct.:

$$F(\rho) = -E_c(1 - \ln x)x - 6\Phi_e y$$

where:  $x = (\rho/\rho_e)^{\alpha/\beta}$ ;  $y = (\rho/\rho_e)^{\gamma/\beta}$ ;  $\alpha = 3(\Omega B/E_c)^{1/2}$ ;  $\gamma = 15(\Omega G/\beta E_{UF})$

---



---

Atom	Inputs					Model parameters				
	$\Omega$	$E_c$	$E_{UF}$	$\Omega B$	$\Omega G$	$f_e$	$\Phi_e$	$\alpha$	$\beta$	$\gamma$
Cu	11.81	3.54	1.30	10.17	4.05	0.30	0.59	5.09	5.85	8.00
Ag	17.10	2.85	1.10	11.10	3.61	0.17	0.48	5.92	5.96	8.26
Au	16.98	3.93	0.90	17.70	3.29	0.23	0.65	6.37	6.67	8.20
Ni	10.90	4.45	1.70	12.28	6.45	0.41	0.74	4.98	6.41	8.86
Pd	14.72	3.91	1.54	17.92	4.99	0.27	0.65	6.42	5.91	8.23
Pt	15.06	5.77	1.60	26.60	6.12	0.38	0.95	6.44	6.69	8.57

# Recommended CDIO project potential functionals

## Lennard – Jones (LJ)

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

- well suited for inert gases (metals), mostly for gas-metals interactions.

## Values of LJ parameters

$\varepsilon$  [ $10^{-16}$  erg];  $\sigma$  [Å]

	<b>He</b>	<b>Ne</b>	<b>Ar</b>	<b>Kr</b>	<b>Xe</b>	<b>Ni</b>	<b>Ag</b>	<b>Rh</b>
$\varepsilon$	14	50	167	225	320	0.66 [eV]	0.34 [eV]	0.67 [eV]
$\sigma$	2.56	2.74	3.40	3.65	3.98	2.24	2.65	2.46

## Morse potential

$$\Phi(r) = D\left(e^{-2\alpha(r-r_0)} - 2e^{-\alpha(r-r_0)}\right)$$

- $D$ ,  $r_0$ ,  $\alpha$  determined by fitting to reproduce bulk properties (lattice constant, cohesive energy, compressibility).
- used mostly for gas-metals (metals) interactions.

### Values of parameters for Morse potential

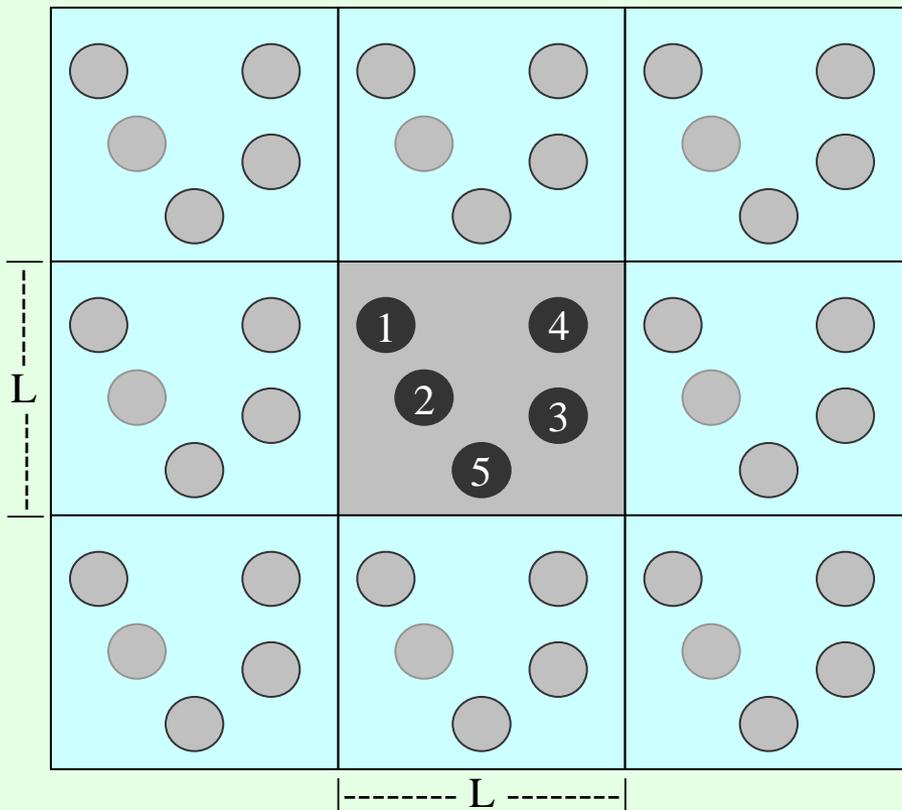
$D$  [eV];  $r_0$  [Å];  $\alpha$  [Å<sup>-1</sup>]

	<b>D</b>	<b>r<sub>0</sub></b>	<b>α</b>
<b>Ag</b>	0.325	3.13	1.36
<b>Rh</b>	0.665	2.87	1.54

## Periodic Boundary Conditions (PBC)

PBC mimic the presence of infinite bulk surrounding the initial N particle system. Main purpose is to eliminate surface effects, which equates to:

small, finite systems → small, infinitely periodic systems



PBCs for two-dimensional system

$$\Phi_{\text{tot}} = \frac{1}{2} \sum_{i,j,\mathbf{n}} V(|\mathbf{r}_{ij} + \mathbf{nL}|),$$

where  $\mathbf{n}$  is number of images of principal simulation cell.

2-dim:  $8 \times \mathbf{n} \times N$  of atoms

3-dim:  $26 \times \mathbf{n} \times N$  of atoms

$N$  can be  $10^3$ ,  $10^4$  or  $10^6$  atoms

**Obs!** PBCs inhibit the occurrence of **long-wavelength** fluctuations, must change  $L$  accordingly.

## Potential Truncation

Based on definition/formulation, potentials have a finite range of interaction.

Typically, **short-ranged** potentials are truncated at  $r_c = \text{cut-off}$ , corresponding to maximum interaction range. Long-ranged potentials are also truncated, and distant contributions (eg. Coulomb interactions) are added separately.

MC simulations: **simple truncation** 
$$V^{\text{trunc}}(\mathbf{r}) = \begin{cases} V(\mathbf{r}) & r \leq r_c \\ 0 & r > r_c \end{cases}$$

MD simulations: **truncation and shift** 
$$V^{\text{tr-sh}}(\mathbf{r}) = \begin{cases} V(\mathbf{r}) - V(r_c) & r \leq r_c \\ 0 & r > r_c \end{cases}$$

Long range corrections are usually not necessary, but one can always add the tail contribution of the potential if required.

**PBC and truncation** are used in conjunction with **minimum image criterion**.



# Monte Carlo simulations

- \* Purely stochastic method.
- \* Quantities of interest are statistical-mechanical ensemble averages.
- \* Example: configurational properties of N-body systems:

$$\langle A \rangle = \frac{1}{Z} \int \dots \int \exp[-\beta U(\mathbf{r}^N)] A(\mathbf{r}^N) d\mathbf{r}_1 \dots d\mathbf{r}_N$$

where  $\beta = 1/KT$ ;  $Z = \int \dots \int \exp[-\beta U(\mathbf{r}^N)] d\mathbf{r}_1 \dots d\mathbf{r}_N$  - configurational integral.

- \* Evaluations are impossible with typical integration algorithms (Simson's rule).

Sample mean integration

$$Z_{NVT} = \frac{V^N}{\tau_{\max}} \sum_{\tau=1}^{\tau_{\max}} \exp[-\beta U(\tau)]$$

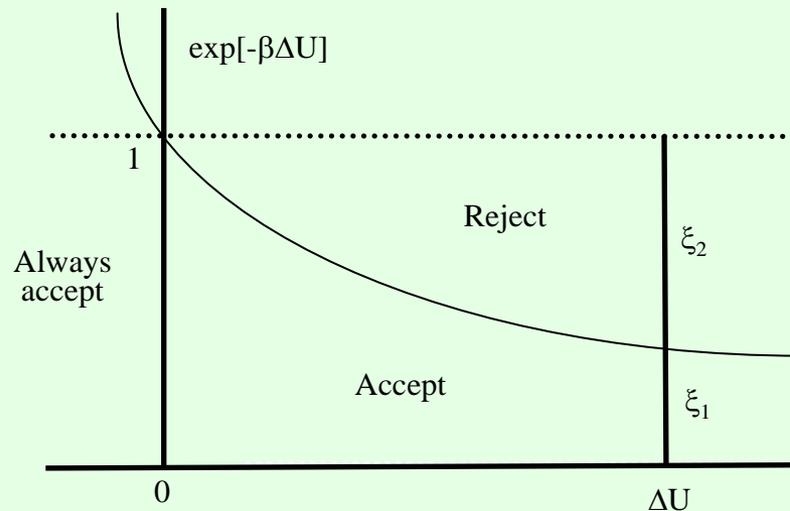
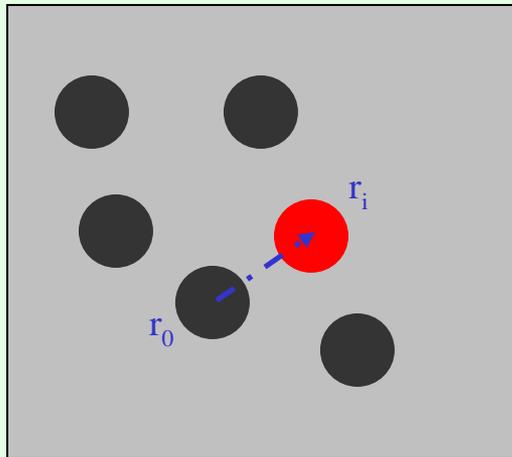
Importance sampling.

$$\langle A \rangle_{NVT} = \langle A \rangle_{\text{trials}}$$

- \* Metropolis [1953] applied the techniques to Markov Chains → Monte Carlo

## Monte Carlo simulations steps:

1. **Assign initial position** to particles & calculate  $U$ .
2. **Move one particle randomly** & calculate new  $U'$  and  $\Delta U = U' - U$ .
3. If  $\Delta U < 0$  - **accept** move.
4. If  $\Delta U > 0$  - **accept** move if  $\xi < \exp[-\beta\Delta U]$ ;  $\xi \in (0,1)$  – random number.
5. If **move rejected** - take the old configuration as the new one  
- repeat 2 - 4 procedure for another arbitrarily chosen particle.
6. For each new configuration **evaluate**  $\langle A \rangle$ .
7. **Repeat** the whole procedure a few million times for adequate statistic



## Molecular Dynamics Simulations

- \* Purely deterministic method.
- \* Follow time evolution of systems by solving Newton's equation of motion:

$$m \frac{d^2 \mathbf{r}}{dt^2} = \mathbf{F}(\mathbf{r}) \quad \text{where } \mathbf{F}(\mathbf{r}) = -\nabla V(\mathbf{r})$$

- \* Properties of statistical ensemble are calculated as time averages.
- \* Microcanonical ensemble (N, V, E) is the common choice + other constraints:  
eg. momentum M  $\implies$  (N, V, E, M) = MD ensemble.

- \* For N particles, problem becomes one of solving systems of 6N 1<sup>st</sup> order DE:

$$\begin{cases} \dot{\mathbf{r}}_i &= \mathbf{\bar{v}}_i \\ m\dot{\mathbf{v}}_i &= \mathbf{\bar{F}}_i \end{cases}; \quad \mathbf{F}_i = -\nabla_{\mathbf{r}_{ij}} \sum_{i \neq j} V(\mathbf{r}_{ij}); \quad \nabla_{\mathbf{r}_{ij}} = \hat{\mathbf{i}} \frac{\partial}{\partial x_{ij}} + \hat{\mathbf{j}} \frac{\partial}{\partial y_{ij}} + \hat{\mathbf{k}} \frac{\partial}{\partial z_{ij}}$$

- \* MD algorithm/concept originally introduced by Alder and Wainright in 1957.

## MD integrators

Typical integration algorithms/methods do not work.

We're dealing with systems of  $10^5 - 10^6$  1<sup>st</sup> order DE (6N equations).

Demands increase in complex ensembles such as (N,V,T) or (N,P,T).

Most CPU time is spent in exceptionally demanding force calculations.

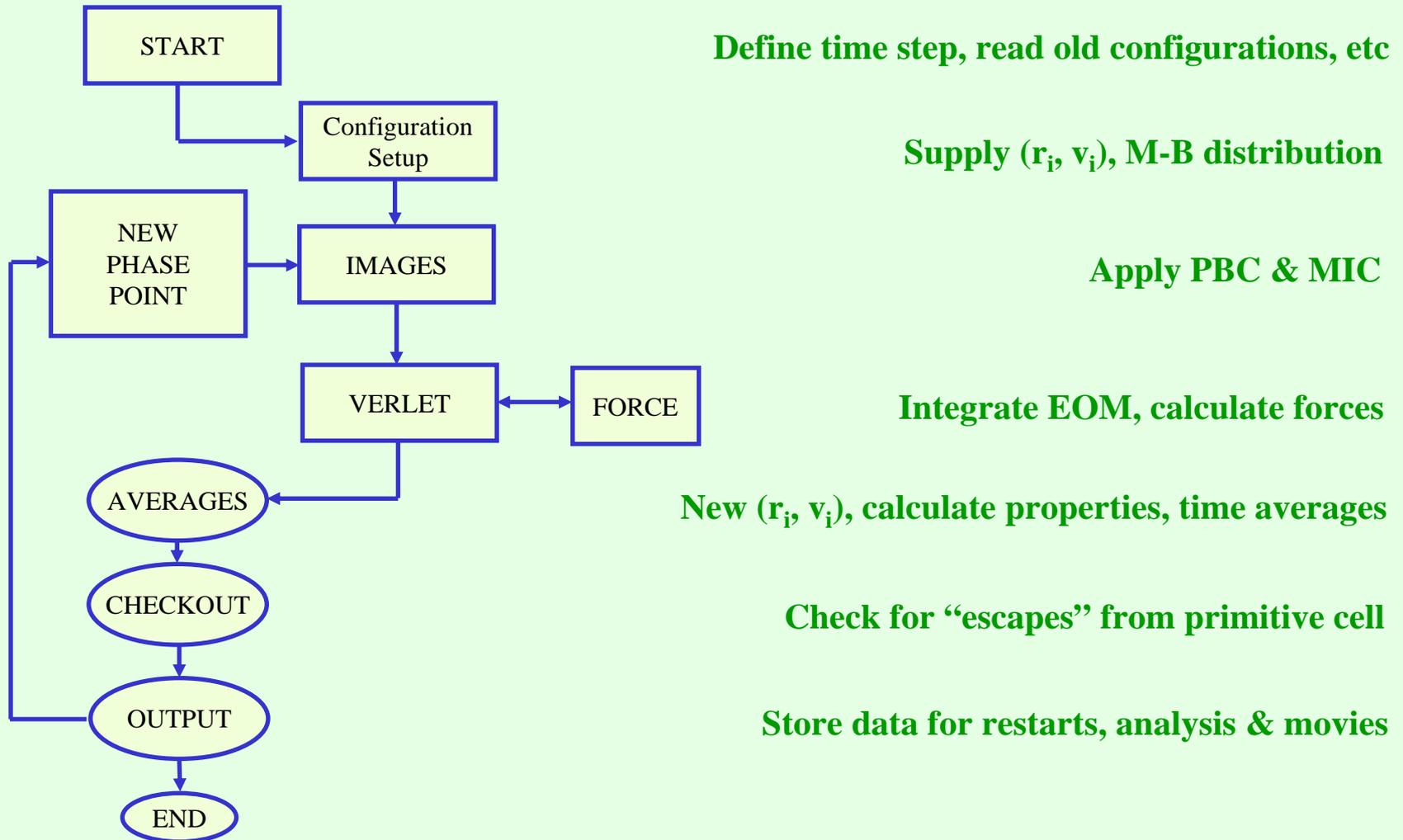
One needs extremely rigorous algorithms.

Selection criteria:

- *simplicity*
- *efficiency*
- *stability*
- *reliability*
- *parallelization.*

**Verlet and Velocity Verlet** algorithms – among the first used and still nearly universal choice for MD simulations, including quantum MD (QMD).

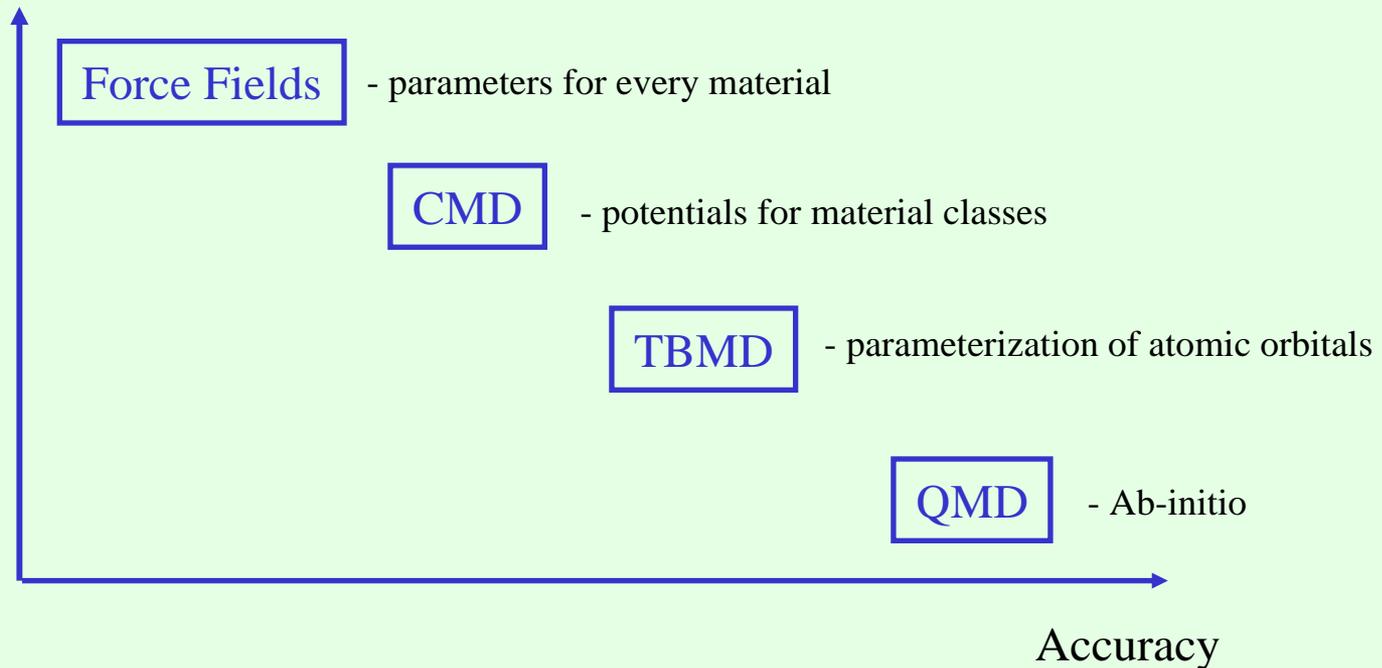
# Molecular Dynamics Flow Diagram



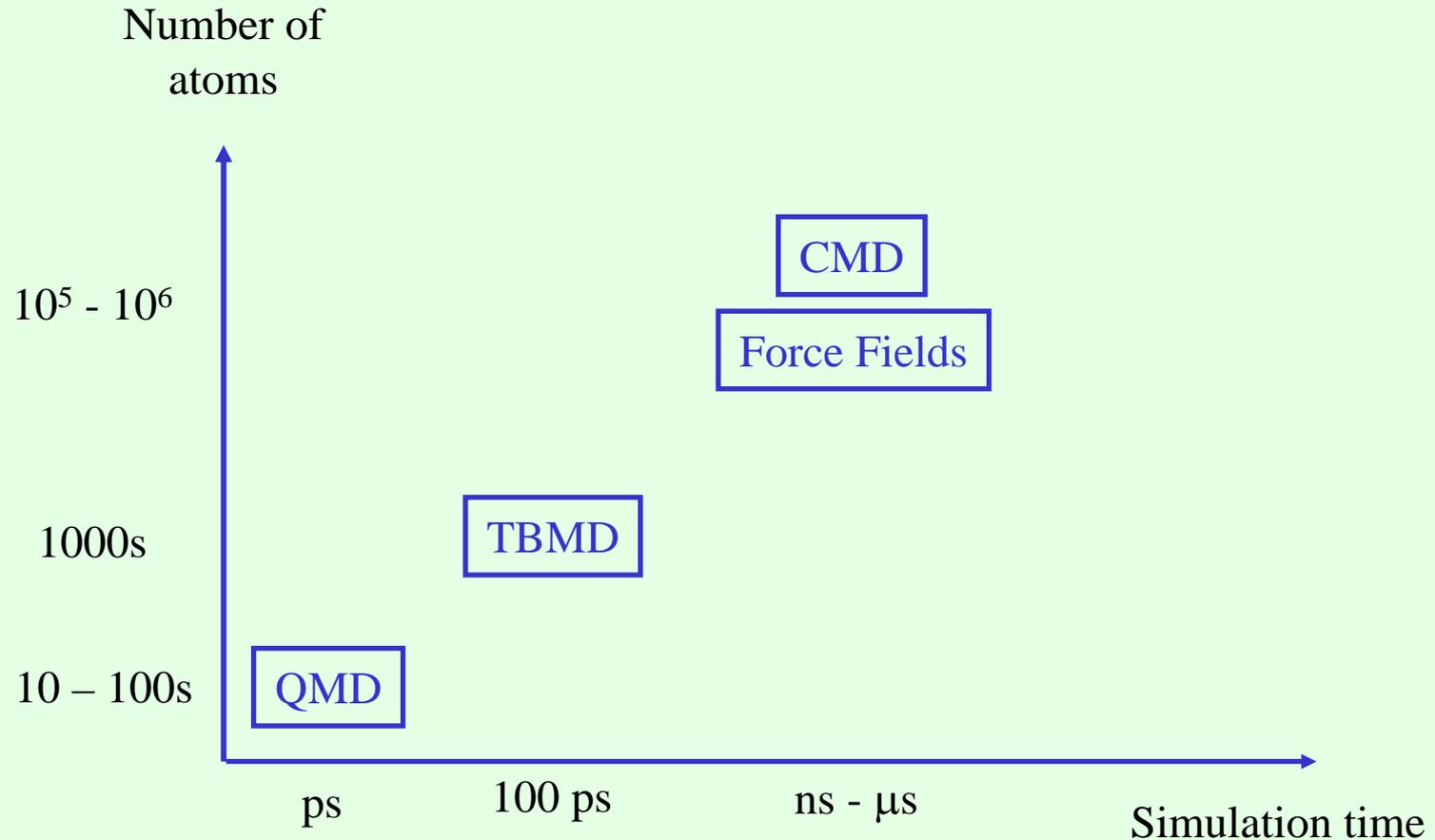
# The Big Picture

We have quantum and classical MC and MD at our disposal. Their applicability to materials science problems depends on a number of factors:

Number of  
parameters

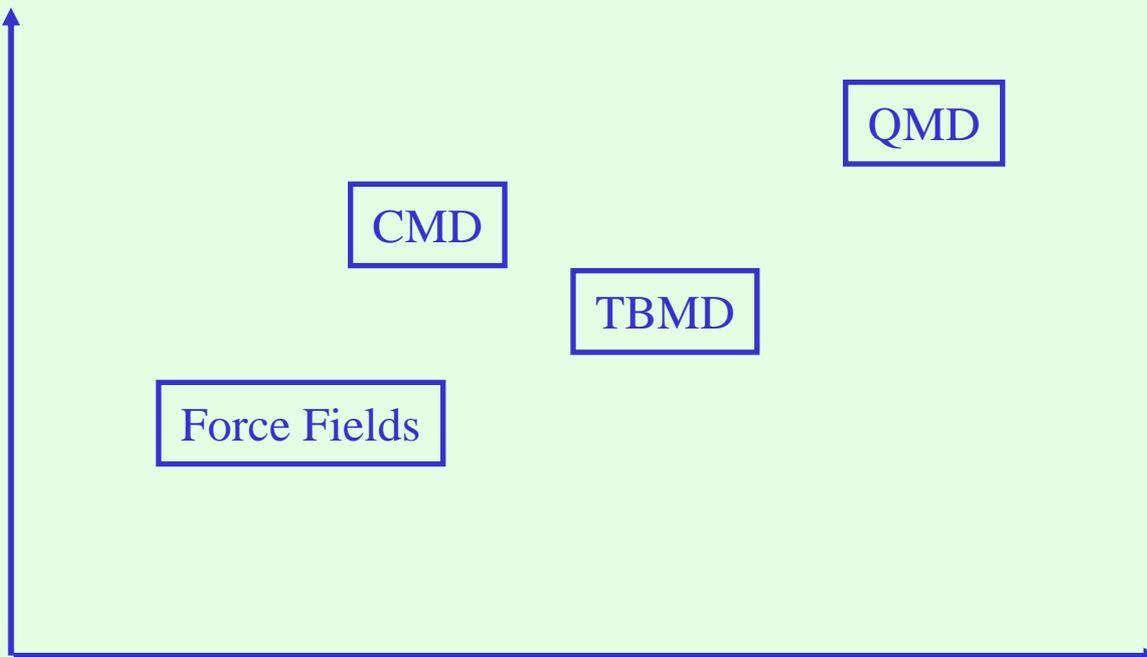


# The Big Picture



# The Big Picture

Versatility



Accuracy

# MC vs MD Comparison

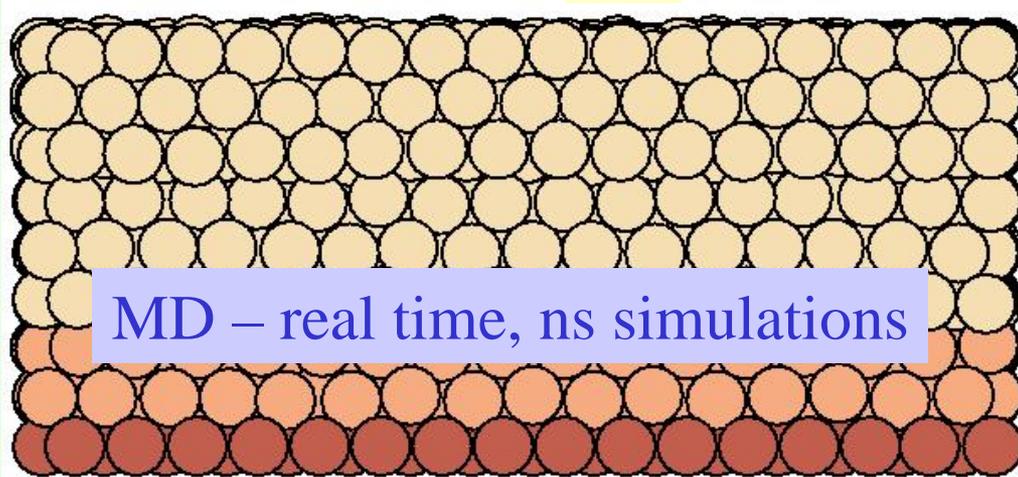
Coated bearings



MC – no time,  $\mu\text{m}$  simulations



MD – real time, ns simulations



## Current & Future Directions

- \* First simulations – 10s of particles.
- \* Today –  $10^6$  particle simulations; smaller systems for 100s of ns.
- \* We have entered the era of applied quantum mechanics.
  - better density-functional-theory (DFT) based (ab-initio) algorithms.
  - one can predict materials properties from information on individual atoms.
- \* DFT codes can study systems of  $\sim 100$ s of atoms for ps in supercomputers.
  - $O(N^3)$  methods  $\implies$  huge research effort for  $O(N)$  techniques.
- \* Future still bright for classical simulations.
  - can simulate systems with  $\sim 10^7$  to  $10^8$  particles
  - alternatively, simulate smaller systems for realistic times ( $\mu$ s, ms)