

Clase 17 (2025)

Dinámica Molecular: Ideas de la técnica

Es muy útil para el estudio de

- Propiedades de equilibrio (termoquímica!)
- Propiedades de transporte (flujos, estados fuera de equilibrio)

* Utiliza la aproximación clásica (leyes de la mecánica clásica)

* Funciona para un amplio rango de materiales y moléculas.

* Sólo cuando se consideran movimientos rotacionales o translacionales de moléculas o átomos muy livianos (Ejemplo: $\text{He}, \text{D}_2, \text{H}_2$) o modos vibracionales con frecuencias ν tales que $\hbar\nu > k_B T$, es necesario considerar los efectos cuánticos.

$$\begin{aligned} V &= V(\vec{r}_i) \\ \rightarrow & f_{ij} = -\frac{\partial V}{\partial \vec{r}_j} \end{aligned}$$

Idea 2

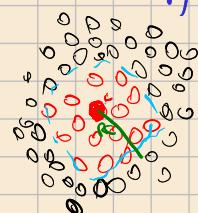
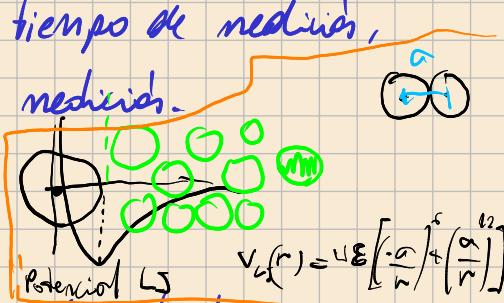
MD es, en muchos aspectos, similar a realizar experimentos reales

Experimento:

- Preparamos la muestra a estudiar
- "Conectamos" la muestra a estudiar a un instrumento de medida (termómetro, viscosímetro, manómetro).
- Medimos propiedades de interés sobre un cieno tiempo.
- Si hay ruido (siempre lo hay!), a mayor tiempo de medida, mejor es el promedio obtenido. Es decir, la medida.

→ MD es similar en esos aspectos!

1) Preparar la muestra: Elegir un sistema modelado de N partículas



modelo → Potenciales de interacción

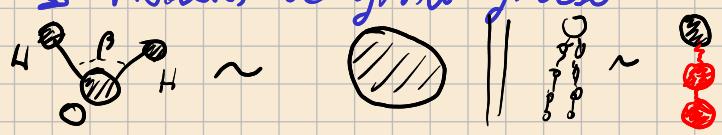
Newton: $m_i \ddot{x}_i = \sum_{j \neq i}^N F_{ij}^c$

$\nabla(r) \rightsquigarrow \bar{F} = \frac{\partial V}{\partial r}$

All atom (metales)

Embedded Atom Model (EAM)

Modelos de grano grueso



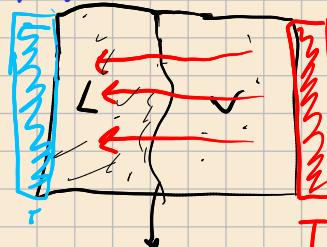
2) Resolver numéricamente las ecuaciones de Newton hasta que no cambien las propiedades con el tiempo (Esto es equilibrio, termalizan, "llegan al nido").



3) Realizar la medición de las propiedades de interés.

* Los errores más comunes también son parecidos a los experimentos:

- Muestra no preparada correctamente.
- Medición de masas/datos corta (error grande).
- No estamos midiendo lo que creemos.



Para medir un observable en MD \rightarrow debemos primero poder expresarlo en términos de posiciones \bar{r}_i y momentos \bar{p}_i ó velocidades.

$$\gamma = \int_0^L C(P_{\text{ext}}, P_{\text{int}}) \, dz$$

Ejemplo: una definición de la temperatura de un sistema clásico many body hace uso del teorema de equipartición de la energía sobre todos los grados de libertad que entran al cuadrado en el Hamiltoniano.

$$E = \underbrace{\sum_{i=1}^N \frac{1}{2} m_i v_i^2}_{\text{Energía cinética}} + \underbrace{\sum_{i,j} V(r_{ij})}_{\text{Energía Potencial}}$$

\rightarrow Energía cinética media por grado de libertad

$$\left\langle \frac{1}{2} m_i v_i^2 \right\rangle = \frac{1}{2} k_B T \quad (1)$$

* Usamos (1) como definición "operacional" de la temperatura

En la práctica: calculamos la energía cinética media total y dividimos por el N° de grados de libertad

$$\sum_{i=1}^N m_i \bar{v}_i = \bar{P}_T = \text{cte. fijo}$$

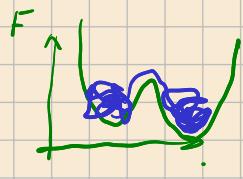
$$T_f = \langle T_{f,i} \rangle N_f = 3N - 3 \quad (\text{sistemas de } N \text{ partículas con } \underline{\text{momento total fijo}}). \quad (\text{tipicamente})$$

Mucho Mucho

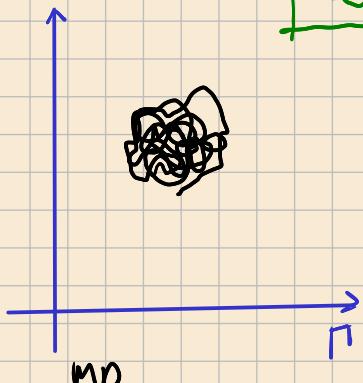
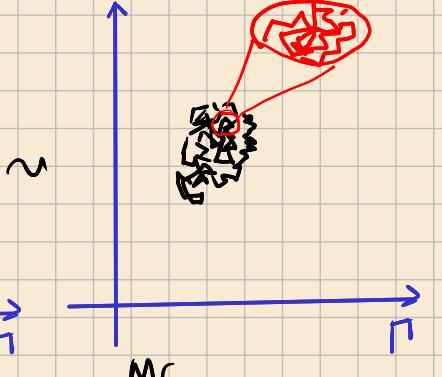
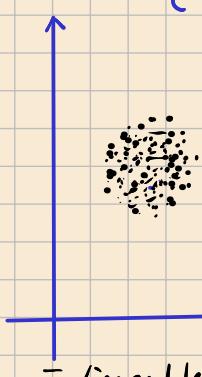
$$T_f(t) = \sum_{i=1}^N \frac{m_i v_i^2(t)}{k_B N_f} \quad (2)$$

* Fluctuaciones relativistas de T $\sim \frac{1}{\sqrt{N_f}}$ $(\text{si } N_f \sim 10^2 - 10^3 \Rightarrow \frac{\Delta T}{T} \sim 5 - 10\%)$

* Comenzamos: ¿por qué funciona MD? Veamos el espacio de fases



Un punto ↓
un estado
del sistema



- Ensemble
- micro Estados que recorre el sistema en una evolución física

- MC
- Cadena de Markov
- muestrean con configuraciones al azar
- Asume ergodicidad

- MD
- trayectoria clásica en el espacio de fases
- Las configuraciones salen de integrar ec's de Newton
- Asume ergodicidad

Programa MD básico

```

loop temporal
program md
    call init
    t=0
    do while (t.lt.tmax)
        → call force(f,en)
        → call integrate(f,en)
        t=t+delt
        call sample
    enddo
    stop
end

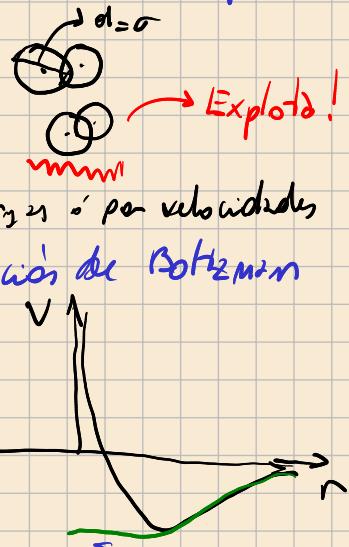
```

Medir variables físicas. Funciones de \bar{r}_i y \bar{v}_i

Bib: Fisica del Sist

Inicialización: Asignamos posiciones ✓ velocidades iniciales

- Los partículas deben tener posiciones compatibles con la estructura que se quiere simular
- Puede ser al azar \rightarrow líquido, gas
- En una estructura cúbica simple (sólido) $\xrightarrow{\text{por fuerza o por velocidad}}$
- Velocidades al azar: puede obtenerse de una distribución de Boltzmann compatible con una $T \rightarrow$ normal o ziggurat



Reescalamiento de velocidades

Puede determinarse las velocidades usando $\text{uni}() \in [-0.5, 0.5]$

$$\text{Tenemos para la "T instantánea"} \quad k_B T(t) = \sum_{i=1}^N \frac{m v_{x,i}^2(t)}{N_f}$$

Temperatura deseada

\Rightarrow Podemos reescalar todas las velocidades con el factor $\sqrt{\frac{T}{T(t)}}$

\Rightarrow Así la primera configuración tiene velocidades tales que la temperatura es la T deseada.

* Puede repetirse para varios pasos de dinámica

Cálculo de las Fuerzas

- Parte más pesada del programa (puede ser $\sim 80\%$ del tiempo o más)

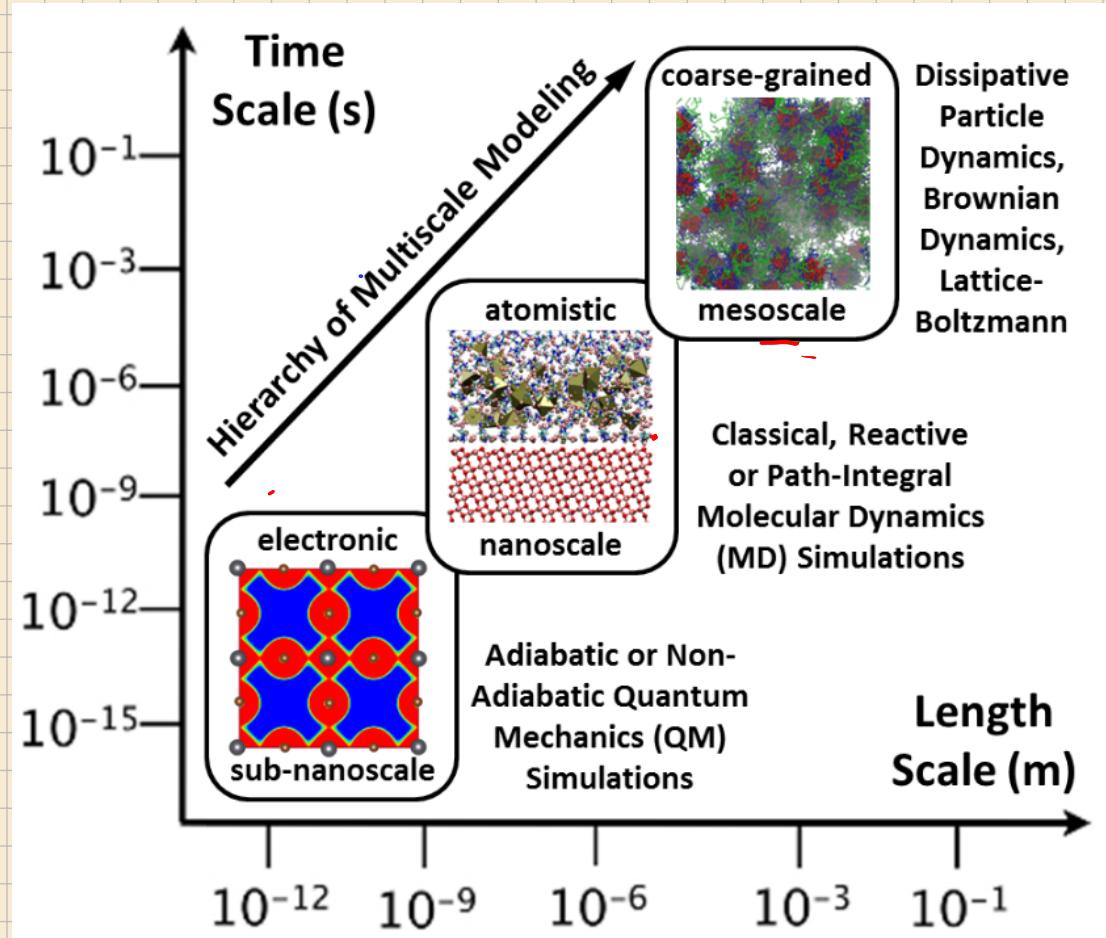
- Si consideramos interacciones aditivas de 2 pares (Ej: Lennard-Jones)

→ Para N partículas debemos evaluar $\frac{N(N-1)}{2} \sim N^2$ distancias entre pares.

→ Sin "trucos" la evaluación de la fuerza es constante $\sim N^2$

* Esto limita el número de partículas (o grados de libertad) que pueden incluirse en una simulación.

Escala y grados de libertad en una simulación



Integración de las Ecuaciones de Movimiento: Algoritmo de Verlet

- * Necesitamos un algoritmo para integrar las ecuaciones de Newton
- * Debe ser rápido y suficientemente preciso. El algoritmo de Verlet, uno de los primeros sugeridos, cumple muy satisfactoriamente ambos requerimientos.

Desarrollo de Taylor en $t + \Delta t$ y $t - \Delta t$.

$$m\ddot{x} = \bar{F} \Rightarrow \frac{\partial^2 x}{\partial t^2} = \frac{F}{m}$$

$$\left\{ \begin{array}{l} r(t + \Delta t) = \underbrace{r(t)}_{\frac{\partial x}{\partial t}} + \underbrace{v(t)\Delta t}_{\frac{f(t)}{2m}\Delta t^2} + \underbrace{\frac{\Delta t^3}{3!} \frac{\partial^3 x}{\partial t^3}}_t + O(\Delta t^4) \\ r(t - \Delta t) = r(t) - \underbrace{v(t)\Delta t}_{\frac{f(t)}{2m}\Delta t^2} - \underbrace{\frac{\Delta t^3}{3!} \frac{\partial^3 x}{\partial t^3}}_t + O(\Delta t^4) \end{array} \right. \quad (1) \quad (2)$$

Sumando (1) y (2):

$$r(t + \Delta t) + r(t - \Delta t) = 2r(t) + \frac{f(t)}{m}\Delta t^2$$

Reescribiendo:

$$r(t + \Delta t) \approx 2r(t) - r(t - \Delta t) + \frac{f(t)\Delta t^2}{m} \quad (3)$$

Algoritmo de posiciones de Verlet

* El error es del orden Δt^2

* No usa velocidades en principio \Rightarrow se reformula como Velocity Verlet para tener expresiones explícitas de las velocidades

Restando (1) y (2):

$$r(t + \Delta t) - r(t - \Delta t) = 2v(t)\Delta t + O(\Delta t^3)$$

$$v(t) = \frac{r(t + \Delta t) - r(t - \Delta t)}{2\Delta t} + O(\Delta t^2) \quad (4)$$

Algoritmo de Verlet . velocidades

Algoritmo Velocity-Verlet

- * Hay varias implementaciones equivalentes del algoritmo de Verlet
 - * Velocity-Verlet es uno de los más utilizados (sí es el que usaremos)
- Forma final del algoritmo Velocity-Verlet:

$$\begin{aligned}
 \xrightarrow{\text{m m}} \quad & r(t+\Delta t) = r(t) + v(t) \Delta t + \frac{f(t)}{2m} \Delta t^2 \\
 & v\left(t + \frac{1}{2}\Delta t\right) = v(t) + \frac{1}{2} \Delta t \frac{f(t)}{m} \\
 & \text{Comptan } f(t+\Delta t) \text{ y } a(t+\Delta t) = \frac{f(t+\Delta t)}{m} \\
 & \underline{v(t+\Delta t)} = v\left(t + \frac{1}{2}\Delta t\right) + \frac{1}{2} \Delta t a(t+\Delta t) \\
 & \quad - \text{compto cantidades físicas}
 \end{aligned}$$

r(t), v(t), f(t) (*)
 Sirve para
 implementación
 y la
 práctica

- * Existen algoritmos de orden superior (implican más tiempo de máquina)
- * Verlet es un algoritmo reversible temporalmente → tiene estabilidad numérica a tiempos largos (ver: Freinkel y Smit 4.3.3 - 4.3.5)

Potenciales de interacción típicos

$$\begin{aligned}
 U(R) = & \sum_{bonds} k_r (r - r_{eq})^2 \\
 & + \sum_{angles} k_\theta (\theta - \theta_{eq})^2 \\
 & + \sum_{dihedrals} k_\phi (1 + \cos[n\phi - \gamma]) \\
 & + \sum_{impropers} k_\omega (\omega - \omega_{eq})^2 \\
 & + \sum_{atoms}^{atoms} \epsilon_{ij} \left[\left(\frac{r_m}{r_{ij}} \right)^{12} - 2 \left(\frac{r_m}{r_{ij}} \right)^6 \right] \\
 & + \sum_{i < j}^{atoms} \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}
 \end{aligned}$$

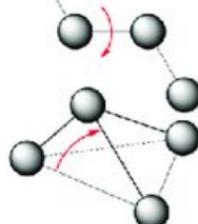
bond



angle



dihedral



improper

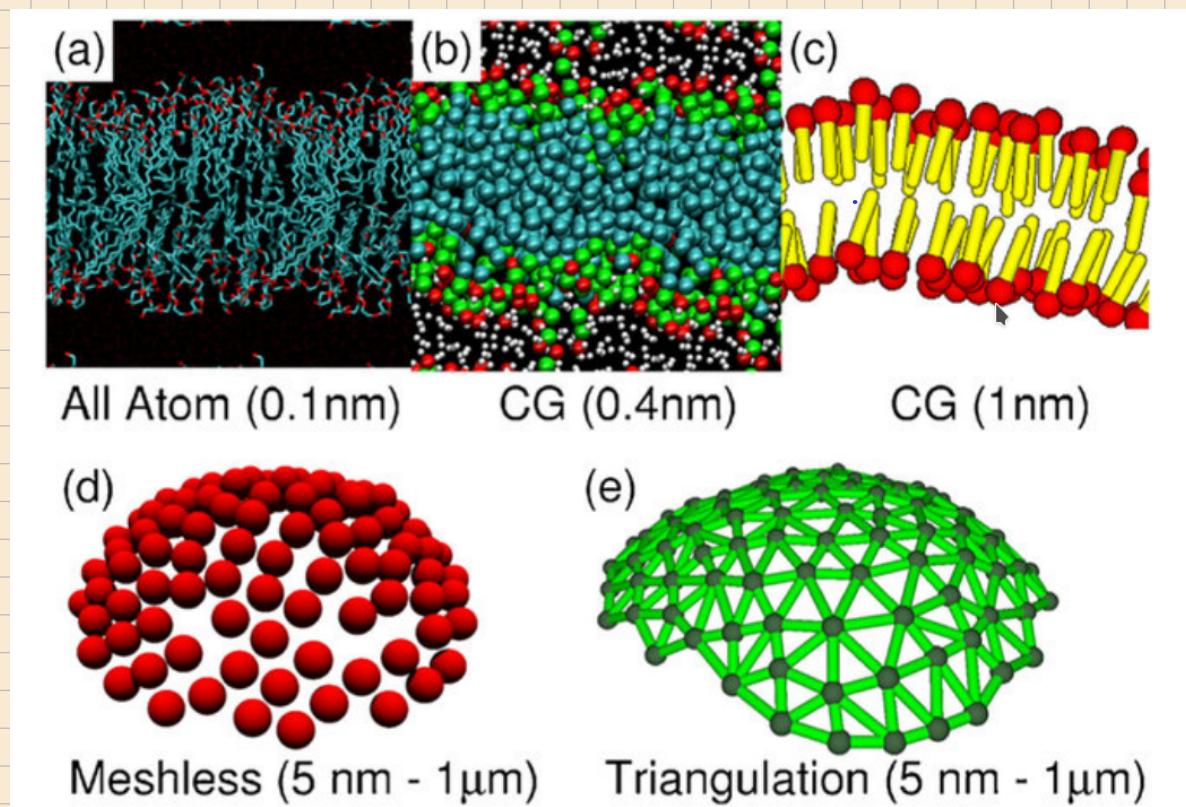
van der Waals



electrostatic



Potenciales de grano grueso



potenciales mejorados por Machine Learning ...

nature
COMMUNICATIONS

ARTICLE

DOI: [10.1038/s41467-018-06169-2](https://doi.org/10.1038/s41467-018-06169-2)

OPEN

Towards exact molecular dynamics simulations with machine-learned force fields

Stefan Chmiela¹, Huziel E. Sauceda², Klaus-Robert Müller^{1,3,4} & Alexandre Tkatchenko⁵

Molecular dynamics (MD) simulations employing classical force fields constitute the cornerstone of contemporary atomistic modeling in chemistry, biology, and materials science. However, the predictive power of these simulations is only as good as the underlying interatomic potential. Classical potentials often fail to faithfully capture key quantum effects in molecules and materials. Here we enable the direct construction of flexible molecular force fields from high-level ab initio calculations by incorporating spatial and temporal physical symmetries into a gradient-domain machine learning (sGMDL) model in an automatic data-driven way. The developed sGMDL approach faithfully reproduces global force fields at quantum-chemical CCSD(T) level of accuracy and allows converged molecular dynamics simulations with fully quantized electrons and nuclei. We present MD simulations, for flexible molecules with up to a few dozen atoms and provide insights into the dynamical behavior of these molecules. Our approach provides the key missing ingredient for achieving spectroscopic accuracy in molecular simulations.

→ Buena modelo para vibraciones moleculares de alta energía.