MD in a Box

Interactive Simulations of Atoms

Bo Jakobsen* & Ulf R. Pedersen**

IMFUFA · Department of Science and Environment (INM) · Roskilde University (RUC)

A Pocket-Sized Physics Lab

"MD in a Box" is a demonstration project porting a browser-based molecular dynamics simulation to a microcontroller you can hold and interact with physically.

The Raspberry Pi Pico 2 costs ~€10 and has a dual core ARM Cortex-M33 CPU — comparable to a mid-1990s Pentium. The entire system including display and accelerometer costs ~€65.

Despite these modest specs, the device runs ~700 MD steps per second with 100 interacting particles.

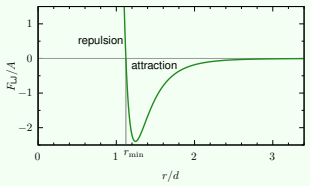
Molecular Dynamics (MD)

Atoms are simulated as particles interacting via pair interactions, following Newton's laws, and with a Langevin thermostat (not included in equation 1).

$$\underbrace{\vec{F}_i = \sum_{j \neq i} \vec{F}_{i,j}}_{\text{Pair interactions}} \qquad \underbrace{\vec{F}_i = m \frac{\mathrm{d}^2 \vec{r}_i}{\mathrm{d}t^2}}_{\text{Newton's 2nd law}} \qquad \underbrace{\vec{F}_{i,j} = -\vec{F}_{j,i}}_{\text{Newton's 3rd law}} \tag{1}$$

Lennard-Jones Force

The Lennard-Jones force is used to model the pair interactions



$$F(r) = A\left[\left(\frac{d}{r}\right)^{13} - \left(\frac{d}{r}\right)^{7}\right] \tag{2}$$

d : particle diameter
A : interaction strength
r^{-13} : short-range repulsion
r^{-7} : van der Waals attraction

Leap Frog Integration

Positions and velocities are found from the forces by numerical integration using the Leap Frog algorithm.

Positions are calculated at t, and velocities at $t + \frac{dt}{2}$ — they "leap" over each other

$$\begin{aligned} x\,(t+dt) &= x\,(t) + v\left(t+\tfrac{dt}{2}\right) \cdot dt \\ v\left(t+\tfrac{3dt}{2}\right) &= v\left(t+\tfrac{dt}{2}\right) + \frac{F\,(x\,(t+dt))}{m} \cdot dt \end{aligned} \tag{3}$$

This half-step staggering makes the integrator *symplectic* — it conserves energy over long simulations.

How to Use It

Button Controls

A	Cycle: Temperature ↔ Gravity
X	Increase parameter
Y	Decrease parameter
B	Hold to exit

Try This

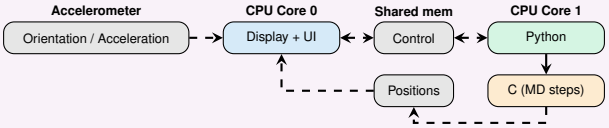
- Heat up → particles move faster
- Cool down → particles move slower, and can crystallise
- Tilt device → gravity follows
- Shake it → particles heat up

MD on a Microcontroller

Hardware:

- Raspberry Pi Pico 2
  - Dual core ARM Cortex-M33 CPU @ 150 MHz
  - 520 kB RAM, single-precision FPU
- 320×240 pixel display
- 9-axis accelerometer

Architecture



Python — C Optimization

MicroPython handles simulation logic and UI, MD calculations are written in C for speedup.

Physical Interaction

9-axis Accelerometer

The accelerometer provides

- Gravity vector
  Tilt the device → gravity rotates with it
- Linear acceleration
  Move the device → box walls move

Performance

Comparing performance of

- **MicroPython + ulab**
- **MicroPython + ulab + C (for MD core)**

	Pico 2		Desktop	
	MD steps / second		MD steps / second	
N	**ulab**	**ulab + C**	**ulab**	**ulab + C**
16	78	12 674	3 894	591 716
25	35	6 816	1 830	425 532
36	18	3 960	903	263 158
64	5	1 553	279	122 699
100	NA	748	NA	68 728
Speedup:	166 – 300 ×		152 – 440 ×	

Particle size: 10 for N < 100, 8 for N = 100.

The physical interaction and screen was not active during tests.

The Pico is roughly a factor 100 slower than a PC, and costs roughly a factor of 100 less.

References & Info

- Browser based MD program: `urp.dk/md`
- "MD in a Box" code: `stuff.upraktisk.dk/md-in-a-box`
- ulab: `https://github.com/v923z/micropython-ulab`
- Hardware:
  - Raspberry Pi Pico 2 (RP2350)
  - Pico Display Pack 2.0 by Pimoroni
  - Fermion: BNO055 Intelligent 9-axis Sensor (Breakout) by DFRobot

* boj@ruc.dk
** urp@ruc.dk