

# Paralelní MD simulace pomocí LJ potenciálu

Lukáš Michalec

*Katedra fyziky, Přírodovědecká fakulta Univerzity J.E. Purkyně v Ústí n.L.  
ročník, specializace*

## Abstract

Seminární práce se zabývá molekulární dynamikou  $N$  částic s Lennard-Jonesovým potenciálem mezi částicemi. Cílem je také implementovat výpočet pomocí více procesů pro zrychlení celkové simulace.

## 1 Úvod

Molekulární dynamika se zabývá pohybem  $N$  částic, které na sebe působí nebo na ně silové působí nějaké vnější pole. V našem případě máme jen vzájemnou interakci Lennard-Jonesovým potenciálem, který má následující tvar:

$$u(r) = 4\epsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right] \quad (1)$$

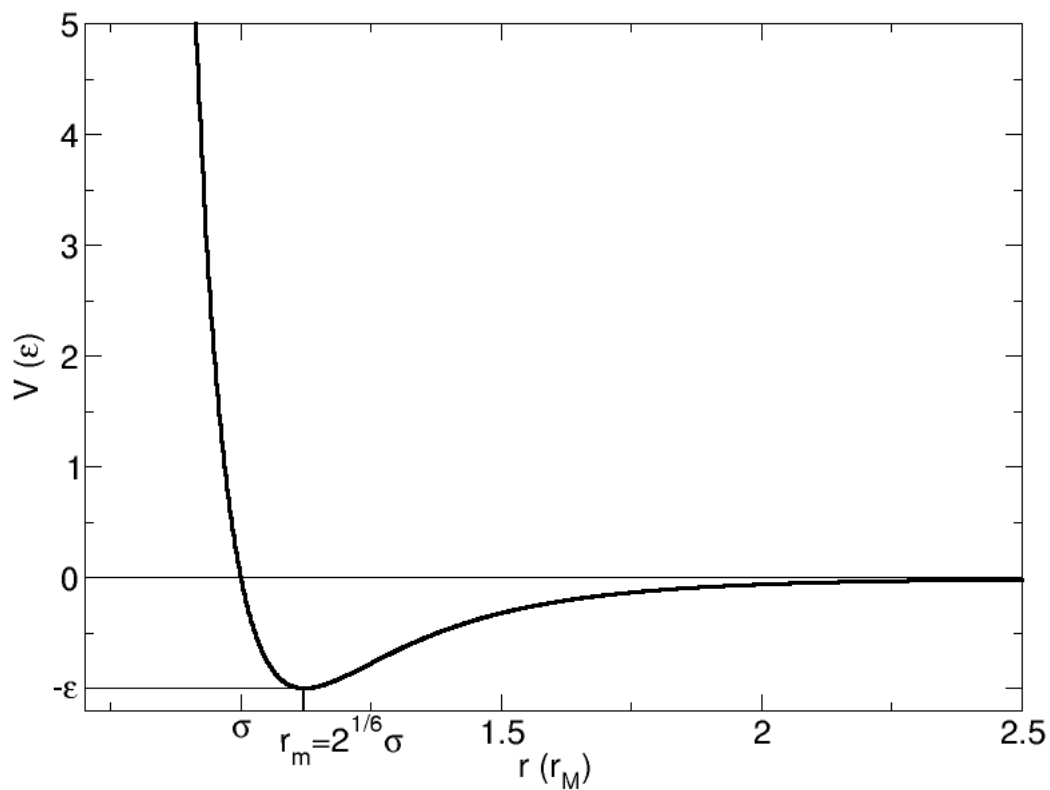


Figure 1: Závislost intenzity na vzdálenosti

Tento potenciál působí na dvě částice, které jsou blízko sebe odpudivou silou a na vzdálenější částice, přitažlivou silou.

Pro výpočet síly, který působí na danou částici, musíme tento potenciál zderivovat a dostaneme:

$$\vec{F}_{ij} = -24\epsilon \left[ 2 \left( \frac{\sigma^{12}}{r_{ij}^{13}} \right) - \left( \frac{\sigma^6}{r_{ij}^7} \right) \right] * \vec{r}_0 \quad (2)$$

Pro celkovou sílu působící na částici, sečteme všechny síly od okolních částic působící na naši částici:

$$\vec{F}_i = \sum_{\substack{j=1 \\ i \neq j}}^N \vec{F}_{ij} \quad (3)$$

Právě tento výpočet se budeme snažit co nejlépe zoptmalizovat a rozparalelizovat.

## 1.1 Pohyb částic

Pohyb částic budeme provádět po výpočtu sil, působící na dané částice. Z vypočtených sil spočítáme zrychlení přes 2. newtonový zákon:

$$\begin{aligned}\vec{F} &= m\vec{a} \\ \vec{a} &= \frac{\vec{F}}{m}\end{aligned}\tag{4}$$

dále pomocí metody **leapfrog**, spočítáme nové polohy:

$$x_{i+1} = x_i + v_i\Delta t + \frac{1}{2}a_i\Delta t^2\tag{5}$$

a dále nové rychlosti:

$$v_{i+1} = v_i + \frac{1}{2}(a_i + a_{i+1})\Delta t\tag{6}$$

Při pohybu částic, budeme uvažovat s periodickými okrajovými podmínkami. Takže, pokud nějaká částice vylétne za okraj naší pracovní oblasti, jiná částice z protějšího okraje zase přilétne.

## 2 Paralelizace a optimalizace

Paralelizaci budeme provádět pomocí distribuce cyklů. Kód, pro výpočet celkové síly působící na částici vypadá následovně:

```
DO i = 1, n
  DO j = 1, n
    IF (i /= j) THEN
      call LJSila(i, j, F)
      F(i) = F(i) + F
    END IF
  END DO
END DO
```

Pokud si uvědomíme, že částice (i) působí na částici (j) stejně jako částice (j) působí na částici (i), ale směr vektoru síly je opačný, proto můžeme tento kód napsat následovně:

```
DO i = 1, n
  DO j = i+1, n
    call LJSila(i,j,F)
    F(i) = F(i) + F
    F(j) = F(j) - F
  END DO
END DO
```

implementujeme distribuci cyklů mezi procesy:

```
DO i = myrank, n-1, nproc
  DO j = i+1, n
    call LJSila(i,j,F)
    F(i) = F(i) + F
    F(j) = F(j) - F
  END DO
END DO
```

a po výpočtu, všechny síly ze všech procesů sečteme:

```
call MPI_ALLREDUCE(F,FSum,N, MPI_REAL, MPI_SUM, MPI_COMM_WORLD
, ier);
```

### 3 Výsledky a diskuze

Na takovém simulačním prostředí, můžeme měřit změnu celkové kinetické energie částic v závislosti na počtu iterací a dostaneme následující graf:

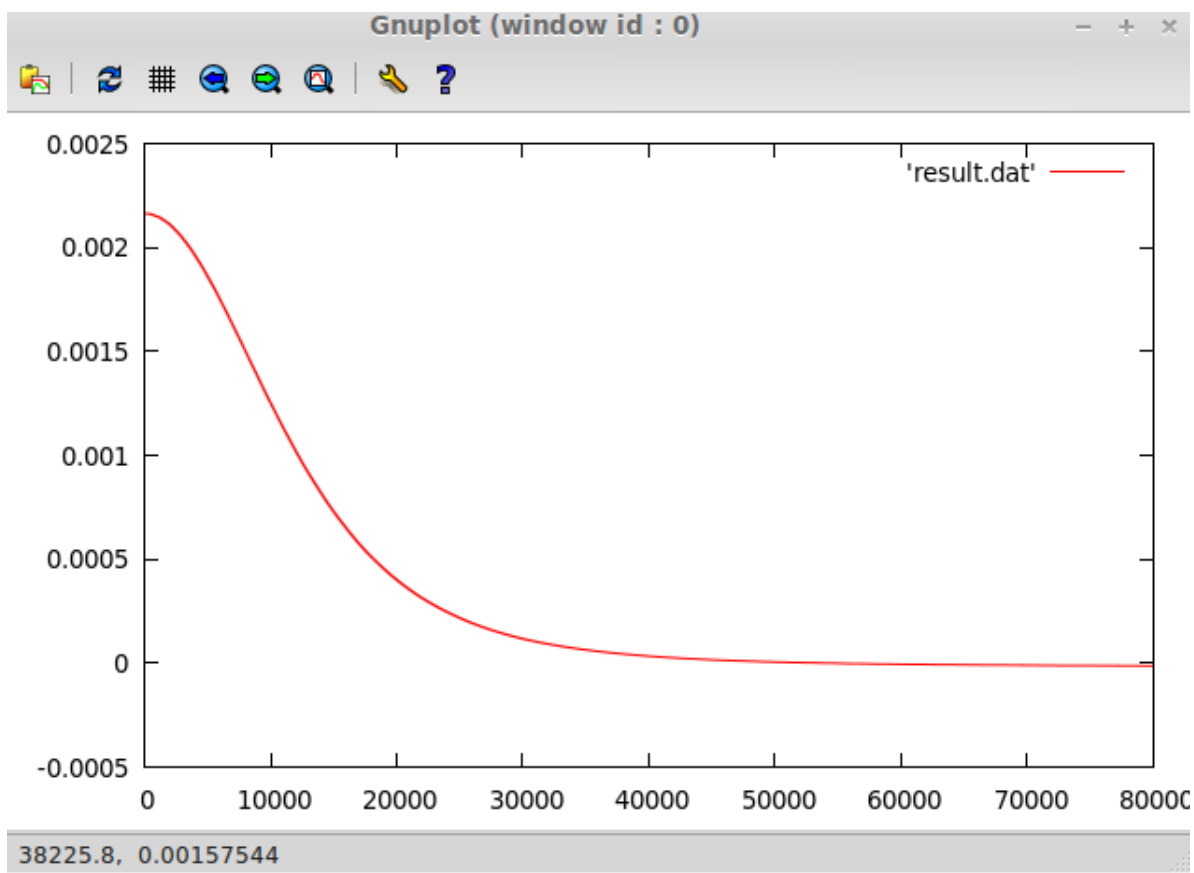


Figure 2: Vývoj celkové energie systému

Celková energie se vyvíjí podle očekávání. Systém se snaží zaujmout stav s nejnižší možnou energií a proto energii klesá. Ten pomalý pokles ze začátku je způsobený tím, že částice po inicializaci mají nulovou rychlost, tak pár iterací trvá, než je ty síly rozhýbou.

## 4 Závěr

Na této úloze jsme si ukázali, jak lze zefektivnit a zrychlit výpočet molekulární dynamiky, ve které se vyskytuje interakce mezi částicemi. A že náš systém se po 60 000 iterací ustálil na nějaké energetické hodnotě.