

Représentation du potentiel de Lennard-Jones

L'utilisation de fonctions en python permet de nombreuses applications par la création de graphiques. En utilisant la "bibliothèque matplotlib/pylab", vous pourrez donc aisément créer des graphes de fonction.

Exemple du [potentiel de Lennard-Jones](#) de l'argon :

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

$$\left[\left(\frac{r_m}{r}\right)^{12} - 2\left(\frac{r_m}{r}\right)^6 \right]$$

où σ est la distance à laquelle le potentiel entre les particules s'annule et ε est l'énergie du puits de potentiel d'interaction. La distance r_m à laquelle le potentiel a cette valeur minimale est reliée à σ par la relation suivante : $r_m = 2^{1/6} \sigma$

```
<sxh python; title : Lennard-Jones-01.py> #!/usr/bin/env python # -*- coding: utf-8 -*- """
Représentation du potentiel de Lennard-Jones Argon :  $\sigma = 3.405 \text{ \AA}$ ,  $\varepsilon/k_B = 118.2 \text{ K}$   $k_B = 1.3806488(13) \times 10^{-23} \text{ J}$  ->  $\varepsilon = 1.632 \cdot 10^{-21} \text{ J}$  """
from pylab import * def f@: sigma=3.405
#angstrom ! epsilon=1.632 # 10-21 J
```

```
s = (sigma/r)**6
s2= 4.*epsilon* (s**2. - s)
return s2
```

```
r=[] u=[] x=3. while x < 10:
```

```
r.append(x)
u.append(f(x))
x=x+0.1
```

```
plot(r, u) show() </sxh>
```

Suggestion : récrire ce programme en utilisant des directives d'importation standard des librairies Matplotlib/NumPy

From:
<https://dvillers.umons.ac.be/wiki/> - **Didier Villers, UMONS - wiki**

Permanent link:
<https://dvillers.umons.ac.be/wiki/teaching:progappchim:lennard-jones?rev=1425400649>

Last update: **2015/03/03 17:37**

