

Sprawozdanie z ćwiczenia z dynamiki molekularnej

Założenia symulacji:

mechanika klasyczna
oddziaływanie siłami Van Der Waalsa

Program do wizualizacji:

jmol

Wybrano język:

C++11

Stworzone struktury danych i funkcje:

vec – abstrakcja reprezentująca trójwymiarowy wektor, przeciążono potrzebne operacje matematyczne: dodawania, odejmowania, mnożenia przez skalar, obliczania Euklidesowej długości między dwoma wektorami jak i jednego.
sign – mała struktura do przechowywania znaku przy generowaniu pędów.

Struktura programu:

argon.hpp – przechowuje struktury i parametry symulacji jak stałe fizyczne, wektory komórki elementarnej, liczbę symulowanych atomów, parametry wirtualnego ciśnienia symulowanego przez sferę otaczającą kryształ (działającą relatywnie dużą siłą skierowaną do wewnątrz), funkcje wypisującą wektor

argon.cpp:

inicjalizacja położeń – wykorzystuje komórkę elementarną, generuje plik tekstowy jak i listę zawierającą początkowe położenia wszystkich atomów

append() - dodaje aktualny stan do pliku zawierającego wcześniejsze położenia

inicjalizacja pędów – zgodnie ze wzorem z instrukcji generuje energie i znaki pędów, normuje energie, do oczekiwanej wartości teoretycznej, liczy i normuje pędy, żeby całkowity kryształ był równy zero, zwraca plik tekstowy i listę pędów.

VDW(), BALL() - przyjmuje położenia zwraca pojedyncze siły van der Waalsa i od otaczającej sfery.

Ep_vdw, Ep_ball, Ek - zwraca energie potencjalną van der waalsa, od sfery i kinetyczną dla pojedynczego atomu.

Ep_tot, Ek_tot, E_tot – zwraca energie potencjalną, kinetyczną i całkowitą całego układu.

calc_T() - liczy temperaturę w kelwinach dla całego układu

calc_force() - bierze położenia i zwraca siły dla każdego atomu.

time_step() - przyjmuje referencje i liczy kolejny krok czasowy.

Main:

rand(12345) – daje stałą losowość pomaga w szukaniu błędów

Program generuje pliki symulacji i grafy energii z temperaturą dla określonego zakresu temperatur z zadany skokiem. Czas każdej symulacji to 1ps, przyjmuje odpowiedni krok czasowy i zapisuje do pliku z energią i symulacją co stałą wielokrotność kroku.

Generacja wykresów:

Wykorzystano notebooki pythonowe i bibliotekę matplotlib, automatyczne wykrywanie zawartości folderów i odczyt parametrów z nazwy.

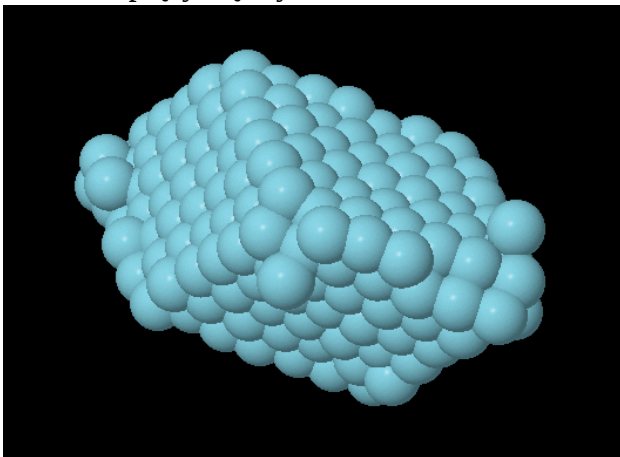
Wnioski i obserwacje:

Dla 0K podczas symulacji obserwujemy ruch i temperaturę układu większą od 0, wiąże się to z brakiem równowagi sił dla wszystkich atomów.

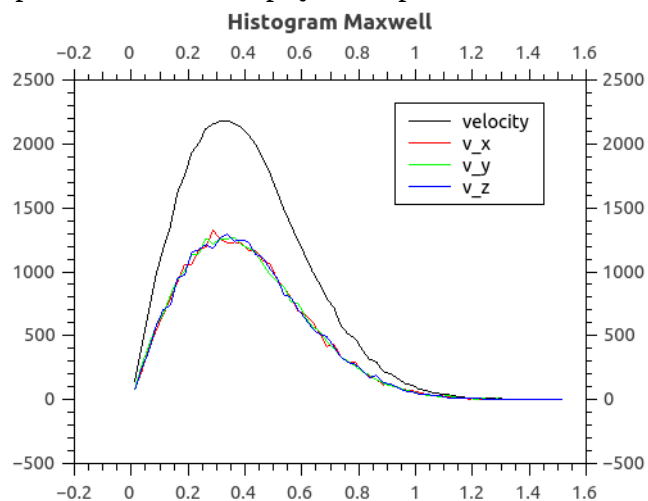
Pomimo, że jest to prosty model, po analizie symulacji można stwierdzić, temperatura topnienia kryształu argonu mieści się w zakresie 70-90K, im więcej symulowanych cząsteczek tym precyzyjniej można ją określić, za tablicami temperatura wynosi 189,36 °C czyli 83.8K.

Zatem nasz model daje przewidywania zgodne z rzeczywistością.

Topiący się kryształ $N=343$ $T_0=80K$



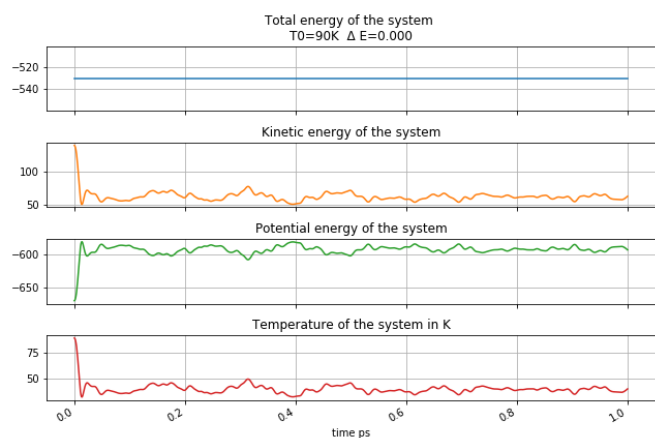
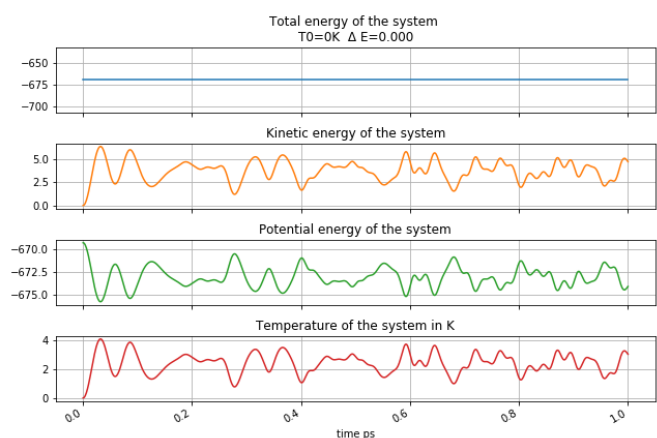
Poprawność rozkładu prędkości potwierdza ich histogram



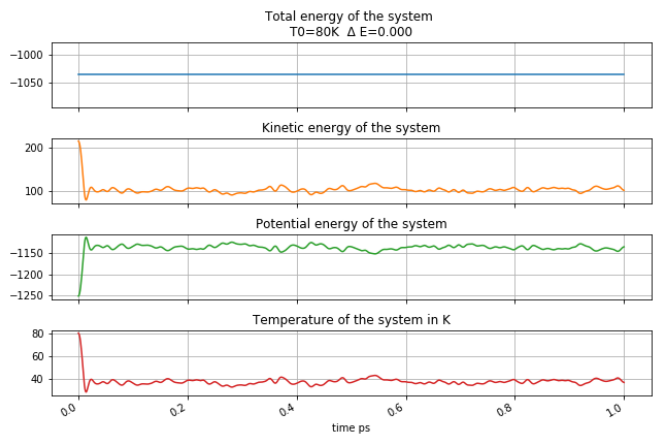
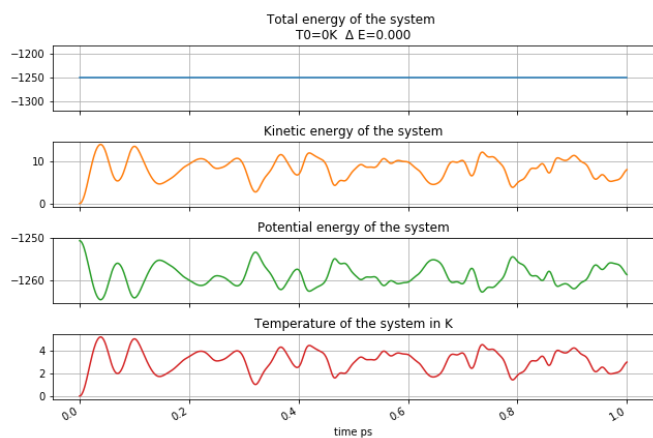
Precyzyjność obliczeń stwierdzono na podstawie zasady zachowania energii, mamy układ izolowany, więc nasza energia całkowita powinna być stała.

Wykonane symulacje spełniają ten warunek do 2 miejsc po przecinku.

Wykresy Układ N=125



Układ N=216



Układ N=343

