

Symulacja fluktuacji magnetyzacji idealnego paramagnetyka w zewnętrznym polu magnetycznym

Mateusz Gałążyn

20 stycznia 2012

1 Cel projektu

Celem projektu jest symulacja fluktuacji idealnego paramagnetyka w zewnętrznym polu magnetycznym opisanego za pomocą modelu Isinga. W tym celu została napisana aplikacja w Javie ilustrująca rozkład spinów w materiale magnetycznym oraz skrypt w Octave, który generuje wykresy zależności fluktuacji (względnej i bezwzględnej) magnetyzacji, a także samej magnetyzacji dla próbek o wielkości $N = 200$ oraz $N = 2000$ spinów.

2 Opis teoretyczny

Energia jednowymiarowego łańcucha spinów w modelu Isinga jest określona zależnością [1]:

$$E = - \sum_{i=1}^{N-1} J s_i s_{i+1} - H M$$

Magnetyzacja materiału wynosi:

$$M = \gamma \sum_{i=1}^N s_i = \gamma N \langle s \rangle \quad (1)$$

gdzie J - oddziaływanie między węzłami, s_i - spin i -tej cząstki w łańcuchu $s_i \in \{-1, 1\}$, γ - współczynnik proporcjonalności oddziaływania jednego elementu łańcucha z polem magnetycznym H , N - ilość cząstek w łańcuchu, $\langle s \rangle$ - wartość średnia spinu w łańcuchu. W przypadku idealnego paramagnetyka mamy do czynienia z brakiem oddziaływania pomiędzy poszczególnymi spinami, więc współczynnik $J = 0$. Dla paramagnetyka energia oddziaływania pojedynczej cząstki wynosi $E_i = -\gamma H s_i$, a dla substancji wyraża się wzorem:

$$E = \sum_{i=1}^N E_i = -\gamma H \sum_{i=1}^N s_i = -\gamma H N \langle s \rangle$$

Rozkład prawdopodobieństwa wystąpienia danego spinu dla cząstki jest rozkładem kanonicznym. Jednocząstkowa suma statystyczna wynosi[1]:

$$z_i = e^{-\beta\gamma H} + e^{\beta\gamma H} = 2 \cosh(\beta\gamma H); \quad \beta = \frac{1}{k_B T}$$

Prawdopodobieństwo wystąpienia spinu w łańcuchu:

$$p(s_i) = \begin{cases} \alpha & = e^{\beta\gamma H}/z_i, & \text{gdy } s_i = +1 \\ 1 - \alpha & = e^{-\beta\gamma H}/z_i, & \text{gdy } s_i = -1 \end{cases} \quad (2)$$

Wartość średnia spinu:

$$\langle s \rangle = p(s_i = 1) - p(s_i = -1) = 2\alpha - 1 = \tanh(\beta\gamma H)$$

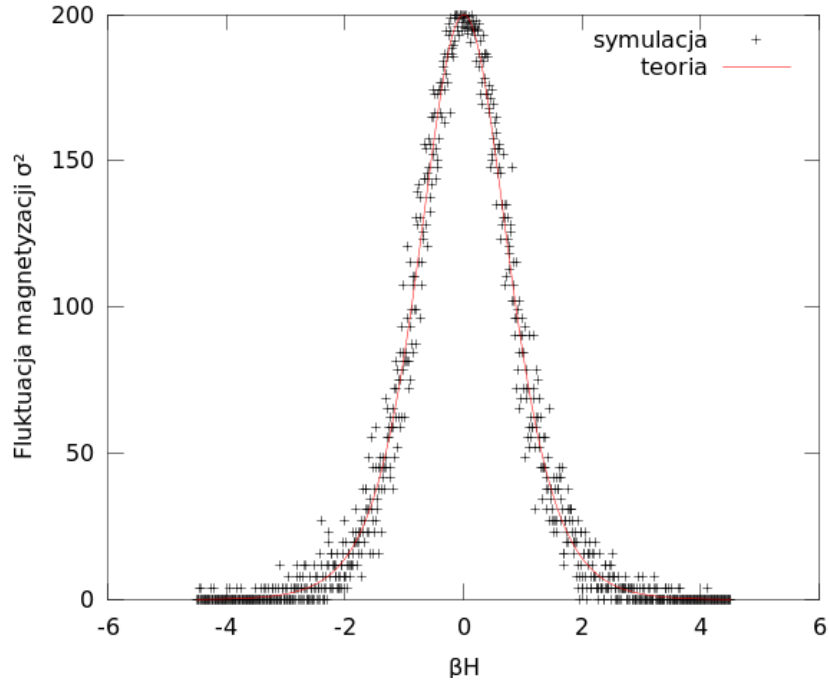
Fluktuacja magnetyzacji jest wyrażona następującą zależnością[2]:

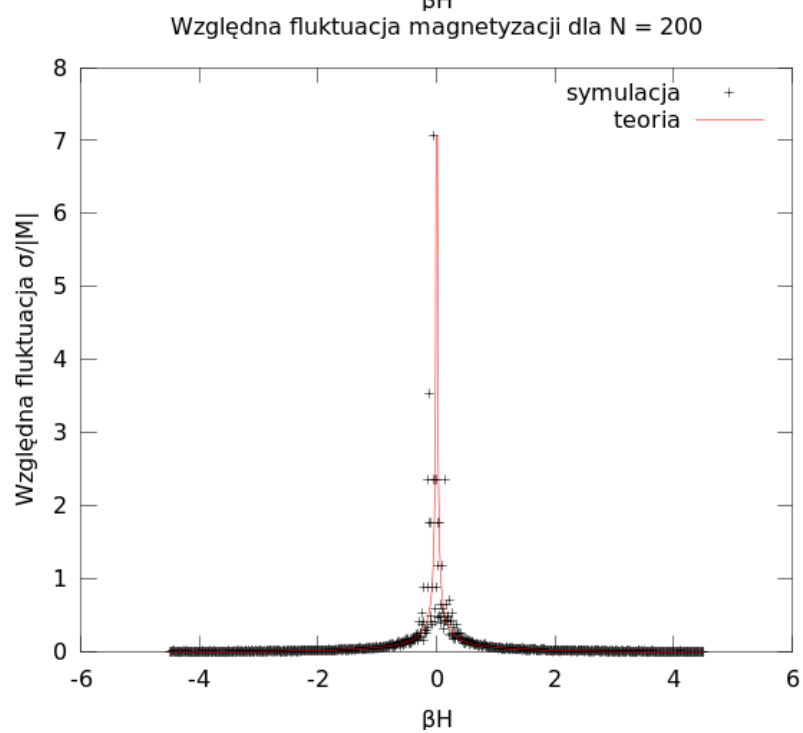
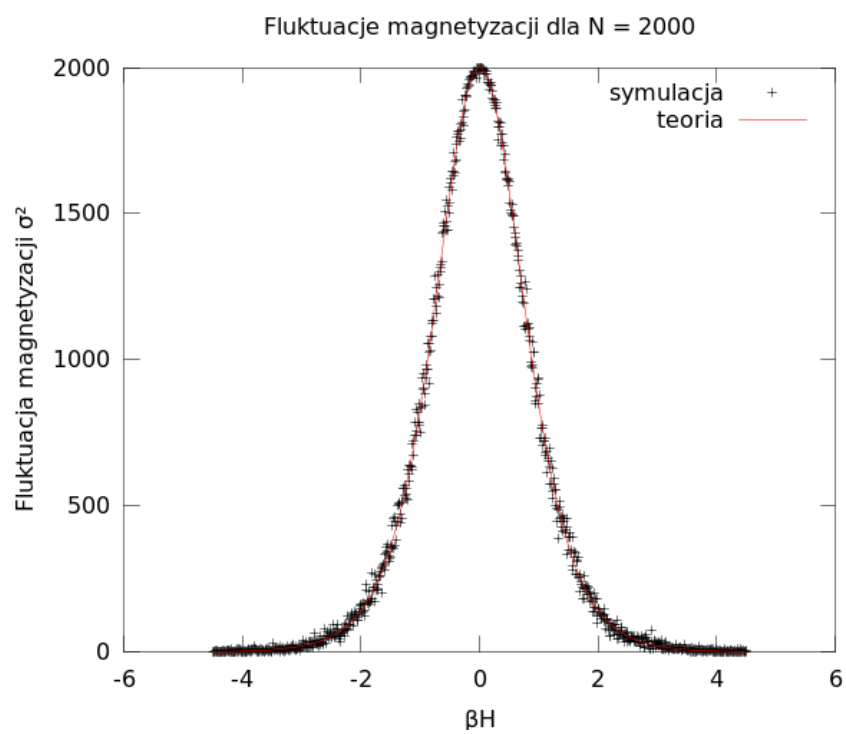
$$\sigma_M^2 = \langle (M - \langle M \rangle)^2 \rangle = \gamma^2 \sum_{i=1}^N \langle (1 - \langle s \rangle)^2 \rangle = \gamma^2 N (1 - \langle s \rangle^2) = \gamma^2 N (1 - \tanh^2(\beta\gamma H)) \quad (3)$$

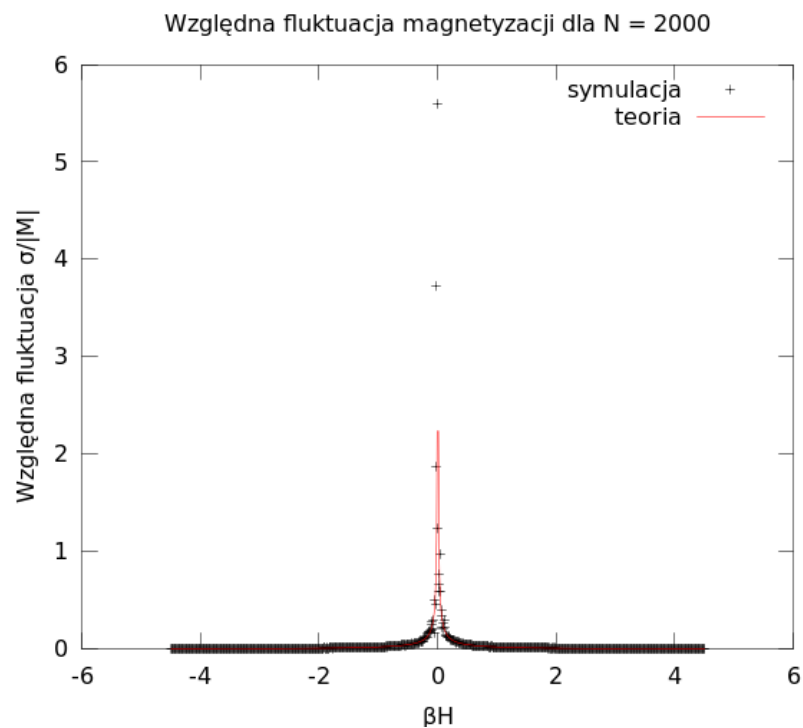
3 Wyniki symulacji

Za pomocą skryptu **simulation.m** zostały wygenerowane wykresy fluktuacji oraz magnetyzacji dla paramagnetyka o ilości spinów $N = 200$ oraz $N = 2000$. Dla uproszczenia sytuacji zostały przyjęte stałe $k_B = \gamma = 1$.

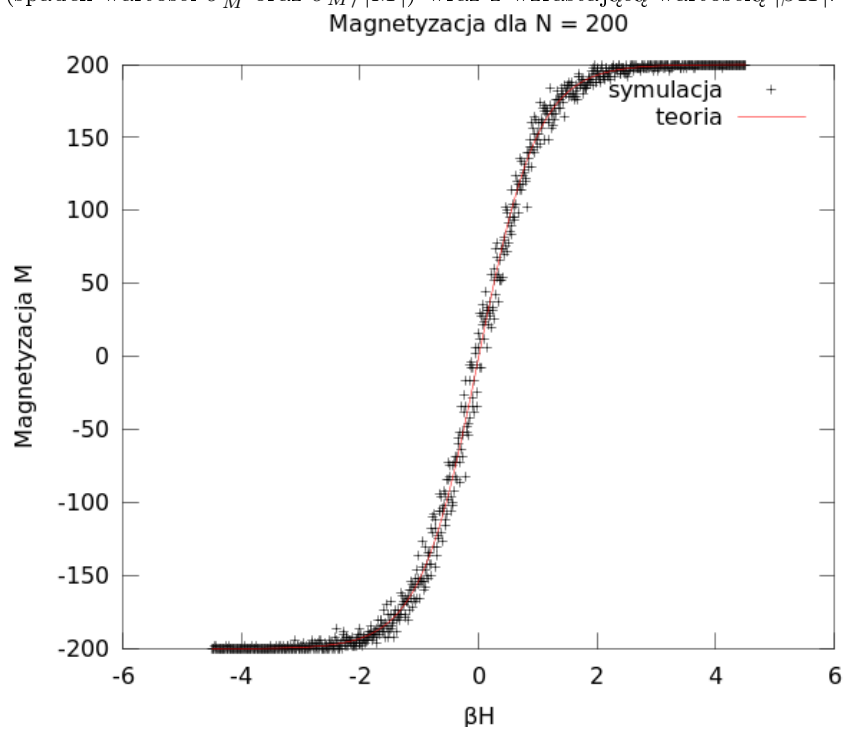
Fluktuacje magnetyzacji dla $N = 200$

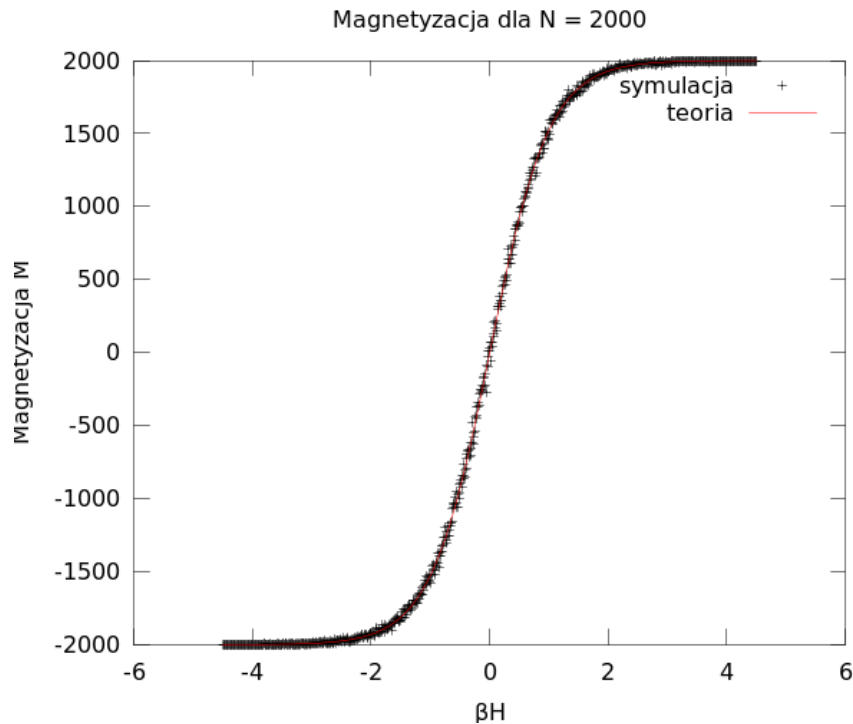






Na wykresach fluktuacji jest zauważalne tworzenie się porządku w sieci spinów (spadek wartości σ_M^2 oraz $\sigma_M/|M|$) wraz z wzrastającą wartością $|\beta H|$.





Zauważalne jest asymptotyczne zbieganie magnetyzacji do określonych wartości dla dużych natężeń pola magnetycznego - następuje wtedy uporządkowanie większości spinów w jednym kierunku. Dla małych natężeń pola magnetycznego wykres magnetyzacji jest prawie liniowy - na tym obszarze jest dobrze spełnione prawo Curie.

Na wszystkich wykresach jest zauważalne zbieganie się wyników z symulacji z krzywymi teoretycznymi wraz z wzrostem liczby spinów.

4 Implementacja

Dołączona aplikacja napisana w Javie symuluje rozkład spinów w idealnym paramagnetyku. Białe kwadraty reprezentują cząstki o spinie równym $s_i = 1$, a czarne odpowiednio $s_i = -1$. Dodatnia wartość pola oraz spinu odpowiada zwrotowi wektora wychodzącego przed płaszczyznę monitora. Dla uproszczenia badanego przypadku zostały przyjęte wartości stałych równe jedności: $k_B = \gamma = 1$. Wartości spinów są generowane za pomocą generatora liczb pseudolosowych z zakresu $[0, 1]$, a następnie wylosowana wartość jest porównywana z parametrem α (2) co pozwala przypisać poszczególnym komórkom na wykresie odpowiadające im wartości spinów. Magnetyzacja i fluktuacja jest obliczana odpowiednio ze wzorów: (1) i (3).

Zawartość załączonych plików źródłowych:

Graph.java - Klasa rysująca wykres.

GraphUpdater.java - Klasa odpowiedzialna za wyznaczenie rozkładu spinów.

ModificationListener.java - Klasa odpowiedzialna za przechwytywanie zdarzeń z interfejsu.

Window.java - Klasa tworząca interfejs.

simulation.m - Skrypt w Octave symulujący zachowanie się paramagnetyka i generujący wykresy fluktuacji.

Literatura

- [1] A. Zagórski: *Fizyka Statystyczna*, OFPW, Warszawa 1994
- [2] A. Fronczak: *Zadania i problemy z rozwiązaniami z termodynamiki i fizyki statystycznej*, OFPW, Warszawa 2006