

# Model Isinga 2D

Tomasz Fał  
t.fas@uw.edu.pl

Celem tego projektu jest zaimplementowanie modelu magnetyzmu Isinga. Jest to model statystyczny badający zachowanie siatki spinów w zadanej temperaturze. W tym projekcie będzie rozważana ewolucja czasowa siatki  $N \times N$  spinów dla zadanych temperatur. Program powinien być w stanie wygenerować losową siatkę spinów, gdzie początkowe prawdopodobieństwo znalezienia danego spinu w stanie '1' lub '-1' jest zadane przez użytkownika. Następnie program wykona ewolucję czasową tej siatki: wyliczy energie poszczególnych spinów biorąc pod uwagę tylko najbliższych sąsiadów, utworzy nowy stan układu poprzez odwrócenie losowego spinu, obliczy prawdopodobieństwo przejścia do nowego stanu i zgodnie z wyznaczonym prawdopodobieństwem przejdzie do nowego stanu lub nie. Po wielu iteracjach układ osiągnie stan równowagi o określonym stopniu namagnesowania (tutaj zdefiniowanym jako średnia wartość spinu siatki). Ostatecznie program powinien wykonać szereg takich ewolucji dla różnych temperatur aby ostatecznie utworzyć wykres zależności średniego spinu od temperatury.

Zwizualizować należy wykres magnetyzacji od temperatury oraz ewolucję siatki spinów w funkcji temperatury. Metoda wizualizacji danych jest dowolna. Można wykorzystać poznana już bibliotekę ImPlot albo wyeksportować dane do pliku i wyrysować je w np. Pythonie.

Model ten opiera się na metodach Monte Carlo, co powinno być względnie proste w pararelizacji. Dlatego też jednym z punktów tego projektu jest poprawne wprowadzenie obliczeń równoległych do tego modelu.

Dodatkowe źródła:

[Model Isinga \(1\)](#) [Model Isinga \(2\)](#) [OpenMP](#)

## Warunki oceniania

- Poprawna implementacja generowania losowej tablicy spinów (5 pkt)
- Poprawna implementacja wyznaczania tablicy energii pojedynczych spinów (15 pkt)
- Poprawna implementacja wyznaczania prawdopodobieństwa zmiany stanu (10 pkt)
- Poprawna implementacja ewolucji układu w czasie (10 pkt)
- Poprawna implementacja wyznaczania namagnesowania w funkcji temperatury (10 pkt)
- Poprawna pararelizacja kodu wyznaczającego namagnesowania w funkcji temperatury (10 pkt)

W projekcie mogą zostać przyznane punkty ujemne za np. brak pliku konfiguracyjnego przy dużej ilości plików źródłowych, popełnianie błędów jak doprowadzenie do wycieku pamięci czy też irracjonalnie nieoptymalny kod.

Przykładowy wygląd programu:

