Wprowadzenie do symulacji i metod Monte Carlo	29.01.2020
Lista nr 6	
Paweł Lorek	

Przypomnijmy krótko **hard-core model** przedstawiony na wykładzie. Mamy dany graf $G = (V, K)$, gdzie $V = \{v_1, \ldots, v_M\}$ (wierzchołki) oraz K jest zbiorem krawędzi. Do każdego wierzchołka przypisujemy *spin* -1 lub $+1$, każde takie przypisanie nazywamy *konfiguracją*.
$\xi \in \{-1, +1\}^V$ jest natomiast *poprawną konfiguracją* (PK) jeśli żadne 2 wierzchołki, które są sąsiadami w grafie G nie mają oba wartości $+1$.

1. Rozważmy rozkład jednostajny na poprawnych konfiguracjach, tj.

$$\pi_G(\xi) = \begin{cases} \dfrac{1}{Z_G} & \text{jeśli } \xi \in PK \\ 0 & \text{w p.p.} \end{cases},$$

gdzie Z_G jest stałą normalizacyjną, tj. $Z_G = \sum_{\xi \in \{0,1\}^V} \mathbf{1}(\xi \in PK)$.

Pokaż jak wygląda łańcuch Markowa o powyższym rozkładzie stacjonarnym stosując algorytm Metropolisa (było na wykładzie, wskazówka: za sąsiadów konfiguracji ξ przyjmyj te konfiguracje, które różnią się wartością w jednym wierzchołku).

2. Rozważmy następujący **uogólniony hard-core model**. Model ten pozwala na różne "intensywności $+1$" w grafie. Jest to zrobione poprzez wprowadzenie parametru $\lambda > 0$. W tym modelu każda poprawna konfiguracja ma prawdopodobieństwo

$$\pi_{G,\lambda}(\xi) = \begin{cases} \dfrac{\lambda^{\eta(\xi)}}{Z_{G,\lambda}} & \text{jeśli } \xi \in PK \\ 0 & \text{w p.p.} \end{cases},$$

gdzie $\eta(\xi)$ jest liczbą "plus jedynek" w konfiguracji ξ, a $Z_{G,\lambda}$ jest stałą normalizacyjną, tj. $Z_{G,\lambda} = \sum_{\xi \in \{0,1\}^V} \lambda^{\eta(\xi)} \mathbf{1}(\xi \in PK)$.

Zastosuj algortym Metropolisa do konstrukcji łańcucha Markowa o powyższym rozkładzie.

3. (Problem komiwojażera). Załóżmy, że mamy daną macierz $\mathbf{M}$ rozmiaru $n \times n$ odległości między n miastami.

Ustalmy kolejność odwiedzania miast: $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n) \in \mathcal{S}_n$ (zbiór wszystkich permutacji zbioru n-elementowego), tj. zaczynamy z miasta σ_1, potem udajemy się do σ_2 itd. z σ_{n-1} do σ_n i z σ_n do σ_1. Całkowita droga jaką przebędziemy to

$$f(\boldsymbol{\sigma}) = \sum_{i=1}^{n-1} \mathbf{M}(\sigma_i, \sigma_{i+1}) + \mathbf{M}(\sigma_n, \sigma_1).$$

Chcielibyśmy znaleźć $\boldsymbol{\sigma}^* = \arg\max_{\sigma}(f(\boldsymbol{\sigma}))$, ale jest to problem *trudny*.

Ustalmy $\beta > 0$ oraz wprowadźmy następujący rozkład na permutacjach:

$$\pi(\boldsymbol{\sigma}) = \frac{1}{Z}\exp(\beta f(\boldsymbol{\sigma})),$$

Zauważmy, że π przyjmuje duże wartości dla "dobrych" permutacji. Losując próbkę z rozkładu π możemy oczekiwać, że znajdziemy dobre przyblizenie $f(\boldsymbol{\sigma}^*)$.

Zastosuj algorytm Metropolisa do konstrukcji łańcucha o powyższym rozkładzie.

Wsk: Dla ustalonej permutacji $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ za "sąsiada" można przyjąc permutację, która różni się transpozycją :

$$\boldsymbol{\sigma}' = (\sigma_1, \ldots, \sigma_{i-1}, \sigma_j, \sigma_{i+1}, \ldots, , \sigma_{j-1}, \sigma_i, \sigma_{j+1}, \ldots, \sigma_n), \quad i, j \in \{1, \ldots, n\}.$$