

# Model Chandrasekhara / Smoluchowskiego - dowolna kombinacja pudeł

Piotr Piękos

22 sierpnia 2019

W tym dokumencie omówiony jest najogólniejszy przypadek modelu pudełek. Zakładamy tutaj, że przejścia pomiędzy pudełkami tworzą łańcuch Markowa. Dodatkowo wierzchołki  $x$  są opisane dwoma parametrami:

- częstotliwością narodzin w danym wierzchołku  $a_{xN}$
- średnią długością przebywania w danym wierzchołku  $a_{xS}$

W każdym wierzchołku mającym częstotliwość narodzin większą od zera cząstki regularnie się "rodzą". W każdym wierzchołku po wejściu do niego (lub po narodzeniu w nim) losowana jest długość życia z rozkładu wykładniczego o zadanym dla wierzchołka parametrze. Po wyjściu cząstka przechodzi do następnego wierzchołka (lub umiera) zgodnie z macierzą przejść.

W dokumencie przedstawiony jest algorytm do symulacji, opis interfejsu oraz przykładowe ustawienia Łańcuchów wraz z trajektoriami.

## 1 Interfejs

Do opisu środowiska potrzebne są dwa pliki:

- nodes - opisujący intensywność narodzin i długość życia we wierzchołku.
- transitions - opisujący macierz przejść po wyjściu z wierzchołka

Format:

nodes jest to plik składający się z dwóch kolumn oddzielonych liczbami. Numer wiersza odpowiada numerowi wierzchołka (licząc od 1). Pierwsza kolumna to intensywność narodzin, druga to długość życia.

transitions to surowy plik z macierzą przejść, kolumny oddzielone są spacjami. **Pierwsza kolumna jest wyróżniona i oznacza przejście do wyróżnionego wierzchołka (śmierć)**

Przykładowe środowiska są zawarte w folderach

- kolko
- koniczynka

## 2 Opis algorytmu

Wynikiem algorytmu będzie tabela ze zdarzeniami, gdzie zdarzeniem jest na przykład narodzenie cząsteczki, albo przejście jej z jednego pudła do innego. Oprócz informacji o zdarzeniu wiersz zawiera też czas wystąpienia zdarzenia

Algorytm składa się z dwóch części

1. Narodziny cząsteczek
2. Podróżowanie

W **pierwszej** części generujemy narodzone cząsteczki - wchodzimy do każdego "rodzącego" wierzchołka, symulujemy wszystkie narodziny które będą miały miejsce w danym wierzchołku podczas ustalonego horyzontu czasu.

W **drugiej** części "chodzimy" cząstkami między wierzchołkami. Ustalamy wierzchołek, dla wszystkich cząstek które aktualnie w nim

przebywają losujemy czas życia oraz nowy wierzchołek. Nadpisujemy "obecny stan" i zapisujemy eventy do historii.

Taka realizacja algorytmu pozwala na wektorową realizację losowania czasów życia i przejść próbek co znacząco przyspiesza czas symulacji.

### 3 Algorytm

#### 1. Narodziny cząsteczek

for  $x$  in rodzące wierzchołki:

Gen  $N \sim Poiss(a_{xN}t)$

for  $i = 1$  to  $N$  do Gen  $U_i \sim U(0, t)$

Dodaj pary  $(x, \text{czas powstania})$  do historii oraz aktualnych wydarzeń

"aktualne wydarzenia" oznaczają ostatnie wydarzenie z daną cząstką, jest ono nadpisywane i przechowuje informacje gdzie jest w danym "momencie" (nie jest to jednak robione chronologicznie z czasem)

#### 2. Podróżowanie

$w = 1$ ,  $W$  - ilość wierzchołków

while najmniejszy czas z aktualnych wydarzeń  $< t$ :

weź cząstki aktualnie będące w  $w$ .

Zasymuluj ich czas życia (z parametrem  $a_{wN}$ ) oraz nowy wierzchołek

zaktualizuj aktualne wydarzenia i dodaj zmienione rzeczy do historii.

$w := w \bmod W + 1$  ( $w \neq 0$ , bo 0 to wierzchołek z martwymi cząsteczkami)

historia zawiera wszystkie wydarzenia, teraz wystarczy je posortować i odpowiednio przetransponować żeby mieć ilość cząsteczek w danej chwili

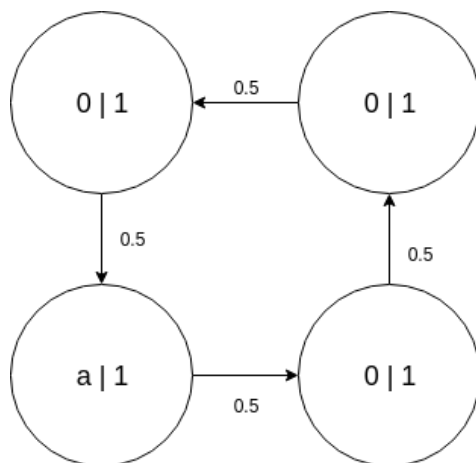
## 4 Przykłady

### 4.1 Przykładowe grafy

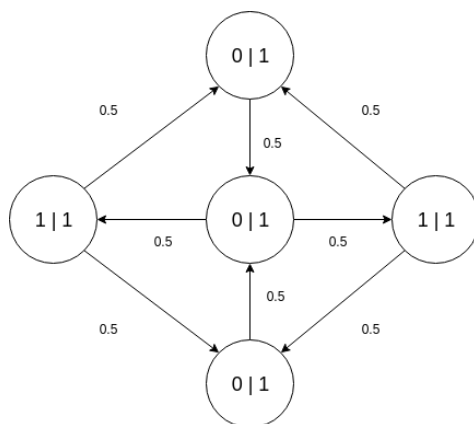
Przykładami są 2 stworzone przeze mnie proste grafy demonstrujące główne funkcjonalności

Oznaczenia:

W wierzchołkach podane są intensywności rodzenia i parametry czasu życia w postaci  $a_N | a_S$  na krawędziach podane są prawdopodobieństwa przejścia. Jeśli wychodzące krawędzie z wierzchołka nie sumują się do 1 to znaczy, że reszta to jest prawdopodobieństwo śmierci (wierzchołek w którym są martwe cząsteczki nie jest explicitnie rysowany)



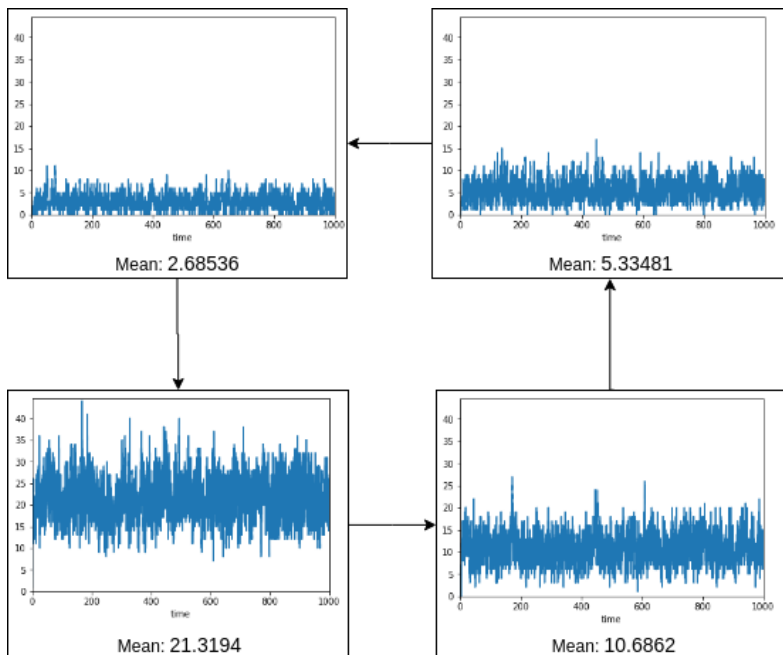
Rysunek 1: **Kolko**. W początkowym wierzchołku (lewym dolnym) cząstki rodzą się z intensywnością  $a$  (przedstawione zostanie kilka przykładów z różną intensywnością) oraz parametr rozkładu wykładniczego z którego losowany jest czas życia wynosi 1. W pozostałych wierzchołkach nigdy się nie rodzi żadna cząstka i parametr długości życia wynosi 1.



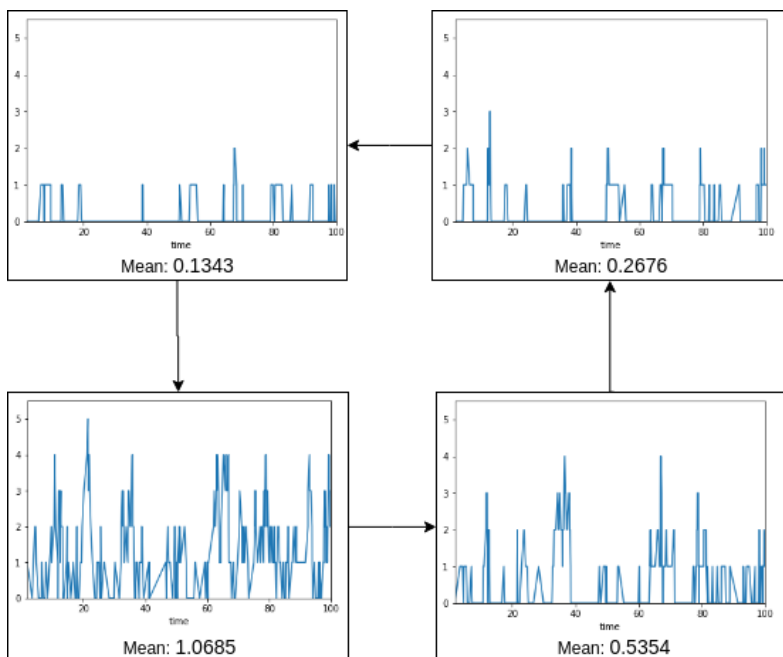
Rysunek 2: **Koniczynka**. Tutaj w dwóch (lewym i prawym) wierzchołkach rodzą się cząsteczki. Mogą umrzeć natomiast jedynie w górnym i dolnym.

Parametry wybrane jako 1 dla prostoty, oczywiście nic nie stoi na przeszkodzie, żeby były dowolne

## 4.2 Kolko

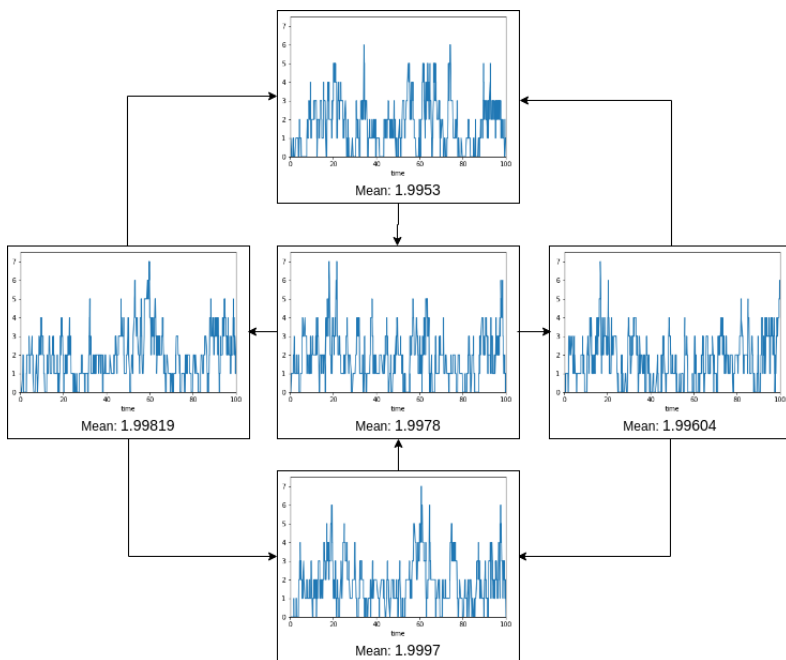


Rysunek 3: **Kolko**,  $a=20$  - Trajektorie każdego wierzchołka. Rysunki przedstawiają trajektorie zawartości każdego z pudełek. Podobnie jak w przypadku jednego lub dwóch pudełek stabilizują się one na pewnym poziomie. Trajektorie przedstawiają symulacje dla  $t = 1000$  średnie były estymowane za pomocą symulacji z  $t = 100000$ . Średnie odpowiadają oczekiwaniom, 20,10,5,2.5 byłyoby bez cyklu - teraz dochodzi część wartości z "niedobitków"



Rysunek 4: **Kolko**,  $\mathbf{a}=1$  - Trajektorie każdego wierzchołka. Podobnie jak na wyższym rysunku tutaj są wykresy trajektorii wartości każdego z pudelek.  $t = 100$  dla wykresów, dla estymacji średniej  $t = 1000000$  (3005897 przejść między wierzchołkami). Dość mocno małe  $a$  oraz  $t$  pozwala zobaczyć i zbadać każde pojedyncze przejście z jednego wierzchołka do drugiego

### 4.3 Koniczynka



Rysunek 5: **Koniczyna** - Trajektorie każdego wierzchołka.  $t = 100$  dla wykresów, dla estymacji średniej  $t = 1000000$  (11989481 przejść między wierzchołkami). Wartość oczekiwana zgadza się z oczekiwaniami, jako że prawdopodobieństwo przejścia wierzchołka przez cały cykl wynosi 50% oraz jest on w pełni symetryczny, jedynie środkowy wierzchołek ma dwie krawędzie dochodzące do niego, ale jest to kompensowane tym, że prawdopodobieństwo dojścia przez nią wynosi 50%

## 5 Uruchamianie symulacji

kod do uruchamiania symulacji - w folderze `wiecej_pudla::`



```
$ python3 sym.py t scheme_name output_file
```

gdzie:

- `t` - czas trwania symulacji
- `scheme_name` - nazwa schematu, folderu w którym są trzymane informacje o schemacie. Na przykład `kolko` albo `koniczynka`.
- `output_file` - nazwa pliku do którego ma się zapisać symulacja (csv)

wykresy zostały stworzone w notebooku (`wiecej_pudel/`)`simulate.ipynb`