



Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych

Grafy i Sieci (GIS)

TEORIA GRAFÓW W MODELOWANIU EPIDEMII

SPRAWOZDANIE 3

Michał Dobrzański
mdobrzezan@mion.elka.pw.edu.pl

Maciej Janusz Krajsman
M.Krajsman@stud.elka.pw.edu.pl

Opiekun projektu:
mgr inż. Łukasz Błaszczuk

15 maja 2017

Spis treści

1	Szczegółowy opis merytoryczny zadania	4
2	Opis algorytmu generowania grafów losowych według modelu sieci Barabásiego-Alberta	4
3	Szczegółowy opis modelu SIS rozwoju epidemii	5
3.1	Założenia modelu SIS	5
3.2	Warunki początkowe	6
3.3	Równania różniczkowe opisujące model	7
4	Opis planu zastosowania grafów losowych do omawianego zagadnienia . .	9
5	Podstawowe założenia implementowanego programu oraz projekt testów .	9
6	Opis kodu	11
7	Instrukcja obsługi	13
8	Wyniki przeprowadzonych badań	15
8.1	Generowanie grafu bezskalowego	15
8.2	Propagacja epidemii	19
8.3	Symulacja dla pierwszej sieci	20
8.4	Symulacja dla drugiej sieci	25
9	Wnioski	26
9.1	Współczynniki	27
9.2	Symulacja grafowa a rozwiązanie teoretyczne	28

1 Szczegółowy opis merytoryczny zadania

Zadaniem projektowym jest **zaimplementowanie modelu SIS** rozwoju epidemii, zbadanie jego właściwości na grafach modelujących populację ludzi, w których wierzchołek odpowiada konkretnej osobie. Należy również porównać otrzymane wyniki z modelem ciągłym dla epidemii opisanym równaniami różniczkowymi.

Zadanie będzie polegało na **wygenerowaniu odpowiednich grafów losowych**, a następnie na uruchomieniu **algorytmu propagacji epidemii** (zgodnego z modelem SIS) dla wierzchołków tych grafów.

Otrzymane wyniki zostaną zestawione z wynikami otrzymanymi za pomocą równań różniczkowych. Dla przejrzystości utworzone zostaną wykresy porównujące oba podejścia. Dobór prezentowanych parametrów dla osi wykresów zostanie określony w trakcie tworzenia projektu.

2 Opis algorytmu generowania grafów losowych według modelu sieci Barabásiego-Alberta

Do zrealizowania projektu potrzebny będzie algorytm generujący grafy losowe według **modelu sieci Barabásiego-Alberta**. Utworzone sieci dzięki temu modelowi są nazywane sieciami przypadkowymi ewoluującymi.

Zaproponowana przez autorów **ewolucja sieci** rozumiana jako zmiany struktury sieci w kolejnych odstępach czasowych modelowana jest poprzez dołączanie nowych węzłów do istniejącej już sieci.

Procedura konstruowania sieci Barabásiego-Alberta obejmuje następujące kroki:

1. Na początkowym etapie ewolucji (czyli w chwili $t = 0$) siecią nazywamy graf pełny (całkowicie połączony klaster węzłów) o rozmiarze: $m_0 \geq 1$. W następnych krokach czasowych $t = 1, 2, 3, \dots$ do sieci dodawane są nowe węzły (jeden węzeł na jeden krok), które tworzą odpowiednio $m \leq m_0$ ($m = \text{const}$) połączeń (czyli krawędzi) do istniejących już wierzchołków sieci.
2. Proces dodawania wierzchołków realizuje **regułę preferencyjnego dołączania**, która mówi o tym, że prawdopodobieństwo, że nowy wierzchołek utworzy połącze-

nie do jednego ze starszych wierzchołków jest wprost proporcjonalne do stopnia wierzchołka starszego.

3. Wzrost sieci kończony jest w dowolnej chwili t . W momencie zakończenia wzrostu sieć ma N wierzchołków (węzłów) oraz E (równanie 1) krawędzi (równanie 2).

$$N = t + m_0 \approx t \quad (1)$$

$$E = mt + \binom{m_0}{2} \approx mt \quad (2)$$

3 Szczegółowy opis modelu SIS rozwoju epidemii

Model SIS opiera się na zestawie równań różniczkowych, które opisują **rozprzestrzanie się chorób zakaźnych**. Służy do określania, czy dana choroba **zaniknie**, czy **ustali się** na konkretnym poziomie. Jest najbardziej ogólnym modelem matematycznym tego zjawiska — nie uwzględnia np. przypadków nosicielstwa i uodpornienia, istnienia osobników w fazie utajonej choroby czy urodzeń chorych. Stan populacji w tym modelu opisany jest **tylko jedną zmienną (I)**, która reprezentuje liczbę osobników zainfekowanych.

3.1 Założenia modelu SIS

- Jednostki mogą znajdować się w jednym z dwóch stanów:

S (*susceptible* — *ang. podatni*) — jednostki są zdrowe, podatne na zakażenie.

I (*infected* — *ang. zainfekowani*) — jednostki są chore, mogą zakażać zdrowe jednostki.

S(t) — liczba osobników podatnych (zdrowych), tj. liczba węzłów w stanie S w danej chwili czasowej t .

I(t) — liczba jednostek zainfekowanych, tj. liczba węzłów w stanie I w danej chwili czasowej t .

- Do zakażenia (zmiany stanu węzła z S na I może dojść na skutek kontaktu jednostki zdrowej i chorej.

N — liczebność populacji, dla której $N = S(t) + I(t)$, Liczebność populacji jest stała, tj. $N = \text{const.}$

β — prawdopodobieństwo, że w pojedynczym kroku czasowym (dt) zdrowy osobnik zostanie zakażony przez chorego sąsiada.

γ — prawdopodobieństwo, że w pojedynczym kroku czasowym (dt) chory osobnik wyzdrowieje, tj. zmieni stan $I \rightarrow S$.

λ — parametr określający **tempo rozprzestrzeniania się epidemii**. Jest to stosunek $\lambda = \beta/\gamma$.

λ_c — **wartość progowa tempa rozprzestrzeniania się epidemii**. Gdy tempo przekroczy tą wartość krytyczną, wówczas epidemia ma szansę stać się powszechną, a badana choroba nabrać charakteru endemicznego. Dla symulacji w czasie nieograniczonym oznacza to endemię.

3.2 Warunki początkowe

Dla sieci losowych oraz bezskalowych (BA) dzięki modelowi SIS można opisać propagację epidemii za pomocą równania różniczkowego określającego tempo zmiany w czasie liczby zakażonych węzłów o zadanym stopniu k . W tym celu należy określić **warunki początkowe** epidemii, czyli w chwili $t = 0$.

Określa się:

$P(k)$ — rozkład stopni wierzchołków otrzymany po wygenerowaniu grafu BA dla wszystkich wartości k .

β, γ — określone prawdopodobieństwa odpowiednio: zdrowy zostanie zakażony przez chorego, chory wyzdrowieje.

$I_k(0)$ — początkowa liczba zainfekowanych węzłów o stopniu k .

$S_k(0)$ — początkowa liczba podatnych węzłów o tym samym stopniu k .

Pytanie, jakie zadaje się przy modelowaniu epidemii najczęściej brzmi: Czy dla zadanych wartości parametrów β i λ oraz dla zadanej początkowej liczby zainfekowanych osobników $i_0(0)$ infekcja rozprzestrzeni się czy nie?

Ponadto dla sieci bezskalowych szuka się wartości parametru λ_c , czyli progu od którego epidemia może stać się powszechna w danej populacji.

3.3 Równania różniczkowe opisujące model

Oprócz wcześniej wymienionych parametrów dla modelu SIS w sieciach bezskalnych określa się prawdopodobieństwo, że dowolna krawędź grafu prowadzi do węzła, który przechowuje stan o wartości „chory”.

$$Q_I = \sum_k Q(k) i_k \quad (3)$$

Gdzie:

k — stopień wierzchołka.

Q_I — prawdopodobieństwo, że dowolna krawędź grafu prowadzi do węzła, który przechowuje stan o wartości „chory”

$Q(k)$ — prawdopodobieństwo, że dowolna krawędź grafu prowadzi do węzła o stopniu k , który przechowuje stan o wartości „chory”.

i_k — prawdopodobieństwo, że węzeł o stopniu k przechowuje stan o wartości „chory”.

Wartość $Q(k)$ można dla danej sieci powiązań dla danej populacji wyliczyć z następującego wzoru:

$$Q(k) = \frac{k}{\langle k \rangle} P(k) \quad (4)$$

Gdzie:

k — stopień wierzchołka.

$\langle k \rangle$ — gęstość sieci. Średni stopień losowo wybranego wierzchołka.

W przypadku sieci BA o tworzeniu połączeń w sieci nie decyduje żadne z góry narzucone prawo, jej **struktura zależy od indywidualnych decyzji jej elementów** (kto kogo poznał, kto się z kim widuje). W rezultacie otrzymujemy **potęgowy rozkład stopni wierzchołków**, w którym w bliskim sąsiedztwie mogą występować wierzchołki o ogromnej liczbie incydentnych krawędzi oraz te niemalże odizolowane od reszty sieci. Ze względu na potęgowy rozkład stopni wierzchołków, wyniki byłyby niewłaściwe. Równania korzystające ze średniego stopnia $\langle k \rangle$ ilustrują dynamikę układu zawierającego same wierzchołki o średnim stopniu. W opisie dynamiki sieci BA należy więc stosować równania dotyczące wierzchołków o **zadanym stopniu k** .

Tempo zmiany osobników zainfekowanych (węzłów o zadanym stopniu k) $I_k(t)$ opisuje się następująco:

$$\frac{dI_k(t)}{dt} = [\beta k Q_I] S_k(t) - \gamma I_k(t) \quad (5)$$

Można również wyliczać tempo zmiany prawdopodobieństwa, że konkretny osobnik (węzeł o zadanym stopniu k) jest zainfekowany:

$$\frac{di_k(t)}{dt} = [\beta k Q_I] s_k(t) - \gamma i_k(t); \quad i_k(t) + s_k(t) = 1 \quad (6)$$

Szczególnie interesujący jest przypadek graniczny, dla $i_k = \lim_{t \rightarrow \infty} i_k(t)$:

$$i_k = \frac{\lambda k Q_I}{1 + \lambda k Q_I} \quad (7)$$

Zgodnie z zapisem w sekcji 3.1, parametr $\lambda = \beta/\gamma$ określa tempo rozprzestrzeniania się epidemii. W powyższym równaniu (7) określone zostało prawdopodobieństwo że węzeł o stopniu k jest w stanie I (jest zakażony) w czasie nieskończonym, czyli po zakończeniu symulacji. Wyzerowanie i_k zależy od Q_I , ponieważ k jest dodatnią liczbą naturalną, a $\lambda > 0$. By wykonywać obliczenia dla wszystkich stopni k , należy zsumować obliczenia dla każdego z nich. Z wzorów (3), (4) i (7) wynika, że:

$$Q_I = \sum_k Q(k) i_k = 1 - \frac{1}{\langle k \rangle} \sum_k P(k) \frac{k}{1 + \lambda k Q_I} = f(Q_I) \quad (8)$$

Gdzie:

k — stopień wierzchołka.

$f(Q_I)$ — funkcja rozkładu prawdopodobieństwa przechowywania stanu „chory” przez węzły o różnych stopniach.

Powyższe równanie może mieć jedno lub dwa rozwiązania. Jednym jest zawsze $Q_I = 0$ (dla $i_k = 0$). Obecność drugiego rozwiązania świadczy o tym, że losowo wybrana krawędź prowadzi do zainfekowanego wierzchołka. Epidemia może się swobodnie rozwijać, więc $\lambda > \lambda_c$. Dzięki temu, warunek:

$$\left. \frac{df(Q_I)}{dQ_I} \right|_{Q_I=0} > 1 \quad (9)$$

umożliwia wyznaczenie λ_c . By funkcja (8) spełniała warunek (9):

$$\left. \frac{df(Q_I)}{dQ_I} \right|_{Q_I=0} = \frac{1}{\langle k \rangle} \sum_k P(k) \frac{\lambda k^2}{(1 + \lambda k Q_I)^2} \Big|_{Q_I=0} = \frac{\lambda}{\langle k \rangle} \sum_k k^2 P(k) = \lambda \frac{k^2}{k} > 1 \quad (10)$$

Więc:

$$\lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle} \quad (11)$$

Okazuje się, że w sieciach bezskalowych wartość parametru λ_c jest bardzo zbliżona do zera. W praktyce powinno to oznaczać, że powstrzymanie rozwoju epidemii jest prawie niewykonalne.

4 Opis planu zastosowania grafów losowych do omawianego zagadnienia

Wygenerowane grafy losowe według modelu Barabasiiego-Arbert posłużą za model badanej populacji. **Wierzchołkami** będziemy reprezentowali poszczególne osoby w danej populacji. **Krawędzie** będą przedstawiały natomiast relacje pomiędzy tymi osobami. Określamy, że dwa wierzchołki są sąsiednie, gdy mają wspólną krawędź grafu.

Siecią losową, bezskalową (w zadanej chwili t utworzoną według modelu BA) będziemy nazywać określoną populację o ustalonej liczności, która zostanie poddana modelowaniu zjawiska propagacji epidemii.

Wierzchołki grafu będą przechowywały informacje o stanie osoby — czy jest ona zdrowa (podatna), czy chora (zakażająca) według modelu SIS. Zostanie zastosowana reprezentacja grafu za pomocą listy wierzchołków wraz z odpowiadającymi im listami sąsiedztwa. Badanie rozwoju epidemii będzie bazowało na tej reprezentacji opisującej stan, w jakim się dany graf znajduje.

5 Podstawowe założenia implementowanego programu oraz projekt testów

Program zostanie zaimplementowany w środowisku Python. Do jego realizacji użyte zostaną biblioteki niezbędne do rysowania wykresów oraz grafów.

Struktura grafu będzie reprezentowana za pomocą listy sąsiedztwa. Dodatkowo w niej dla każdego wierzchołka umieści się dodatkową informację o jego aktualnym stanie (S lub I według modelu SIS). W środowisku Python będzie to słownik tworzony za pomocą konstruktora *dict()*. Badanie algorytmu propagacji epidemii sprowadzi się do przechodzenia po listach sąsiedztwa i zmianach stanów dla wierzchołków.

W implementowanym programie przeprowadzone zostaną następujące testy:

1. Sprawdzenie poprawności algorytmu generującego grafy losowe według modelu sieci Barabasiiego-Alberta:

Zostaną wygenerowane sieci losowe BA (o różnych parametrach m_0 , m oraz t). Ich rozkłady stopni wierzchołków zostaną przedstawione na wykresie. Prawidłowo utworzone sieci według modelu BA powinny charakteryzować się w przybliżeniu **potęgowym rozkładem stopnia wierzchołka**. Ten rozkład przybiera postać liniowego przebiegu na wykresie w skali podwójnie logarytmicznej. Na tym wykresie wartość nachylenia krzywej przekłada się na wartość wykładnika dla rozkładu potęgowego. W programie utworzony zostanie taki wykres i przedstawione na nim będzie kilka przykładowych rozkładów dla różnych sieci losowych BA.

2. Przeprowadzenie symulacji rozwoju epidemii według modelu SIS dla różnych parametrów:

Następnie dla wcześniej wygenerowanych sieci losowych przeprowadzona zostanie symulacja rozwoju epidemii. Dla każdej sieci utworzone będzie kilka różnych warunków początkowych (odpowiednia liczba wierzchołków w stanie S i I w chwili t_0 , różne parametry γ , β). Następnie dla kolejnych chwil $t = 1, 2, 3, \dots$ będzie badany rozwój epidemii. Najistotniejszy jest wynik dla $t \rightarrow \infty$, ponieważ opisuje stan końcowy układu. Wyniki porównane zostaną ze spodziewanym rezultatem otrzymanym z równań opisujących model SIS. Sprawdzone zostanie również, jak model zachowuje się dla $\lambda > \lambda_c$ oraz $\lambda < \lambda_c$ (w tym celu graf nie może być za mały, a wartości λ_c i tak będą bardzo niewielkie, ze względu na potęgowy rozkład stopni wierzchołków grafu).

6 Opis kodu

Zgodnie z założeniami projektu, podstawowym zadaniem była grafowa implementacja modelu SIS. By sprawdzić, czy uzyskane wyniki są poprawne, zaimplementowano też rozwiązanie teoretyczne, na podstawie równań różniczkowych opisujących ten model.

Zastosowane metody:

main() — Główna procedura, staruje kolejnością wykonywanych zadań i parametrami pozostałych metod. Wypisuje część informacji w konsoli.

generate_graph(m_0, m, t) — Generuje graf Barabasiiego-Alberta. Parametry: m_0 — liczba wierzchołków w początkowym grafie pełnym, m — liczba krawędzi łączących każdy nowo dodany wierzchołek z istniejącym wcześniej grafem, t — liczba wierzchołków do dodania w procesie generowania grafu.

pref_addition(g, m) — Dodaje do grafu g nowy wierzchołek, o m krawędziach incydentnych.

get_graph_degree(g, print_deg=False) — Zwraca rozkład stopni wierzchołków grafu g . Parametr *print_deg* określa, czy wektor wynikowy będzie wypisany (*True*) czy nie (*False*).

print_graph(g) — Wypisuje graf g w konsoli, w formacie:

```
wierzcholek : [stan , [ lista polaczen ]]
```

np.

```
1 : [0 , [2 , 3 , 4 , 6 , 7 , 10 , 12 , 20 , 21 , (... ) , 2989]]
```

plot_graph(g, (...)) — Rysuje graf g . Pozostałe parametry określają wygląd grafu, są zgodne z biblioteką *matplotlib*.

average_graph_degree(degrees, n_graphs, m_0, m, t) — Zwraca uśredniony rozkład stopni wierzchołków grafu, na podstawie wygenerowanych w tym celu n_graphs grafów o identycznych parametrach (m_0 , m , t , zgodne z *generate_graph(m_0, m, t)*) jak początkowy graf.

plot_graph_degree(g, m_0, m, t, n_graphs=5, alpha=3, ref_length=0.8)
— Rysuje rozkłady odniesienia oraz rozkład stopni wierzchołków w grafie g

o parametrach (m_0, m, t , zgodnych z $generate_graph(m_0, m, t)$). Parametr n_graphs określa, na podstawie ilu grafów tworzymy uśredniony rozkład. Parametry $alpha$ oraz ref_length służą do sterowania rozkładami odniesienia (pierwszy to współczynnik rozkładu potęgowego, drugi to względna długość rozkładów odniesienia).

propagate_sis(i_k , $beta$, $gamma$, t_max , $degrees$) — Teoretyczna implementacja modelu SIS na podstawie rozkładu prawdopodobieństwa zakażenia i_k , w przedziale czasowym $[0, t_max]$. Parametry $beta$ oraz $gamma$ opisują model i odpowiadają prawdopodobieństwom zakażenia i wyzdrowienia. Wektor $degrees$ zawiera rozkład stopni wierzchołków w modelu grafowym.

infinite_sis(i_k , $beta$, $gamma$, $degrees$) — Oblicza rozwiązanie równania różniczkowego na podstawie rozkładu prawdopodobieństwa zakażenia i_k , dla czasu dążącego do nieskończoności. Parametry $beta$ oraz $gamma$ opisują model i odpowiadają prawdopodobieństwom zakażenia i wyzdrowienia. Wektor $degrees$ zawiera rozkład stopni wierzchołków w modelu grafowym.

simulate_sis($graph$, $beta$, $gamma$, t_max , $degrees$) — Grafowa implementacja modelu SIS dla grafu $graph$, w przedziale czasowym $[0, t_max]$. Parametry $beta$ oraz $gamma$ opisują model i odpowiadają prawdopodobieństwom zakażenia i wyzdrowienia. Wektor $degrees$ zawiera rozkład stopni wierzchołków.

calc_q_i($degrees$, i_k) — Oblicza wartość Q_I na podstawie rozkładu stopni wierzchołków $degrees$ oraz rozkładu prawdopodobieństwa zakażenia i_k

trim_zeros(i_k , $degrees$) — w rozkładzie prawdopodobieństwa zakażenia i_k zastępuje braki (dla rozkładu stopni wierzchołków $degrees$ nie występujących w grafie) wartością -1.

plot_sis(t , $infected_vect_calc$, $infected_vect_sim$, $beta$, $gamma$, k) — Rysuje wykres prawdopodobieństwa infekcji i_k (dla wierzchołków o stopniu k) w zależności od chwili t , zawierający dane pochodzące z grafowej symulacji modelu ($infected_vect_sim$) oraz obliczeń teoretycznych ($infected_vect_calc$) na podstawie równań różniczkowych, opisujących model. Parametry $beta$ oraz $gamma$ opisują model i odpowiadają prawdopodobieństwom zakażenia i wyzdrowienia.

- infect(graph, idx)** — Zakaża celowo wybrany wierzchołek na podstawie indeksu wierzchołka *idx* grafu *graph* (przypisuje mu stan *I*).
- heal_all(graph)** — Leczy całą populację (przypisuje wszystkim wierzchołkom grafu *graph* stan *S*).
- calc_i_k(graph)** — Liczy rozkład prawdopodobieństwa $i_k(t)$, że wierzchołek o stopniu *k* przechowuje stan „chory” (reprezentowany jako wektor) dla grafu *graph* w dalej chwili czasowej *t*.
- toss(prob)** — Losuje liczbę w zakresie 0...1 z rozkładu jednorodnego i zwraca wartość *True*, gdy wylosowana liczba jest większa, niż parametr *prob*. W pozostałych przypadkach zwraca wartość *False*.

7 Instrukcja obsługi

Program sam wykonuje wszystkie niezbędne kroki (opis procedur: sekcja 6):

I) Generowanie grafu

- 1) Wygenerowanie grafu (sekcja 2),
- 2) Wypisanie grafu w konsoli,
- 3) Pobranie rozkładu stopni wierzchołków grafu,
- 4) Rysowanie grafu (graficzne przedstawienie, czytelne tylko dla grafów o nie-dużej liczbie wierzchołków),
- 5) Rysowanie wykresu przedstawiającego rozkład stopni wierzchołków oraz teoretyczne rozkłady odniesienia (sekcja 5),
 - Możliwe narysowanie rozkładu uśrednionego, na podstawie *n_graphs* wygenerowanych (na podstawie identycznych danych) rozkładów.

II) Modelowanie epidemii

- 1) Zainfekowanie wybranego wierzchołka (pacjent zero),
- 2) Określenie początkowego rozkładu prawdopodobieństwa zakażenia (i_k) wierzchołków o stopniu *k*, dla wszystkich stopni wierzchołków występujących w grafie,
- 3) Obliczenie teoretycznego przebiegu propagacji epidemii (sekcja 3, na podstawie równań różniczkowych, opisujących model SIS (dokładniej w sekcji 3.3),

- 4) Obliczenie ww. równań dla $t \rightarrow \infty$ — teoretycznej granicy rozwoju epidemii,
- 5) Przeprowadzenie symulacji przy pomocy wygenerowanych wcześniej grafów bezskalowych (sekcja 4),
- 6) Rysowanie wykresów, przedstawiających przebieg rozwoju epidemii uzyskany przy pomocy algorytmu grafowego, oraz teoretyczny przebieg, wyznaczony przy pomocy równań różniczkowych (sekcja 5),
 - Możliwe narysowanie rozwoju uśrednionego, na podstawie n_sim symulacji (z identycznymi danymi wejściowymi).

Wszystkie te procedury wywoływane są kolejno przez główną funkcję programu, *main()*. Niezbędne parametry znajdują się na początku kodu:

I) Generowanie grafu

- m_0** — liczba węzłów w początkowym grafie pełnym,
- m** — liczba połączeń tworzonych z istniejącym grafem przez nowo dodawany wierzchołek; stopień nowego wierzchołka w chwili dodania,
- t, total_nodes** — liczba wierzchołków do dodania w trakcie tworzenia sieci oraz całkowita ich liczba,
- n_graphs** — liczba grafów do uśredniania wyników rozkładu $P(k)$.

II) Modelowanie epidemii

- beta** — współczynnik zakażenia β ,
- gamma** — współczynnik zdrowienia γ ,
- t_max** — czas trwania symulacji (liczba iteracji infekowania/zdrowienia),
- n_sim** — liczba symulacji do uśrednienia,
- max_plots** — liczba wykresów przedstawiających propagację epidemii (każdy dla konkretnego stopnia wierzchołków, k).

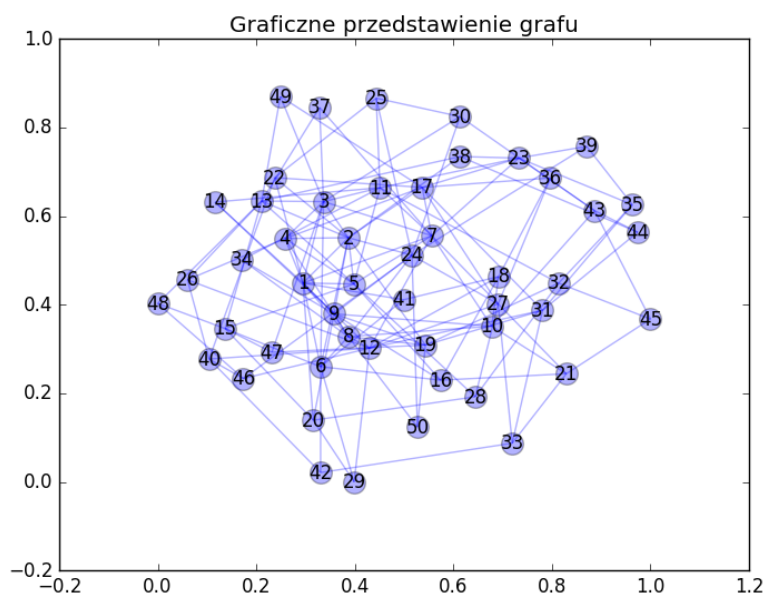
8 Wyniki przeprowadzonych badań

8.1 Generowanie grafu bezskalowego

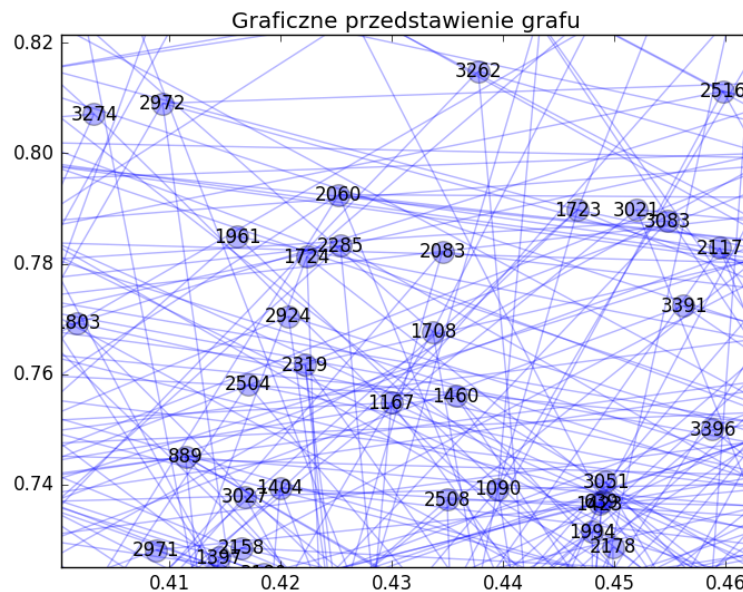
Program poprawnie generuje sieci bezskalowe według modelu BA. Są one reprezentowane za pomocą słownika, gdzie kluczem jest indeks wierzchołka. Natomiast wartość przechowuje stan wierzchołka - „chory” lub „zdrowy” reprezentowany odpowiednio 0 lub 1 i listę sąsiedztwa - indeksy wierzchołków, do których prowadzą krawędzie. Przykładowa realizacja sieci bezskalowej jest przedstawiona na rysunku 1. Tą sieć wygenerowano dla parametrów:

- $m_0 = 3$
- $m = 3$
- $t = 47$

Na rysunku 2 przedstawiono zbliżenie dużej sieci BA o 3400 węzłach. Można wyróżnić centra, tak zwane „**huby**”, czyli nieliczne wierzchołki o bardzo wysokim stopniu w stosunku do innych wierzchołków w grafie.



Rysunek 1: Graf o 50 wierzchołkach



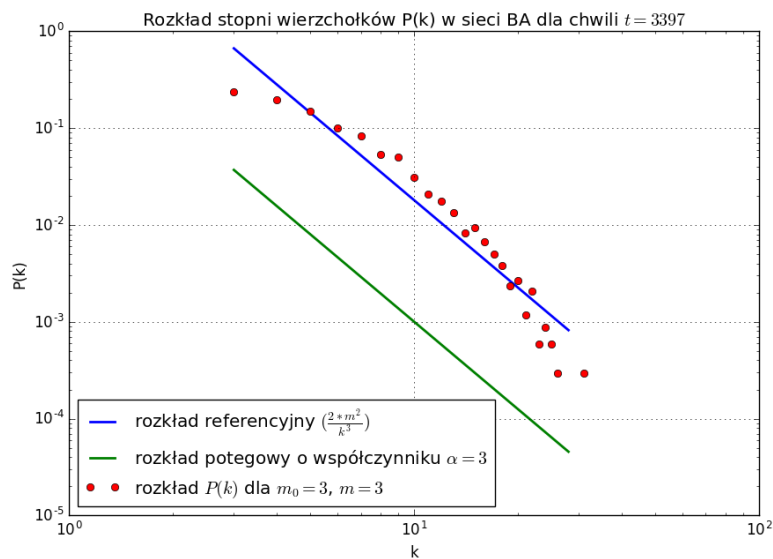
Rysunek 2: Graf o 3400 wierzchołkach — przybliżenie

W celu weryfikacji poprawności generowania sieci BA utworzono **wykresy rozkładów stopni wierzchołków $P(k)$** dla konkretnej chwili czasowej t . Na osi poziomej wykresu oznaczono kolejne wartości stopni wierzchołków. Osi wykresów są przeskalowane do skali **podwójnie logarytmicznej**. Przedstawiono rozkłady referencyjne wyznaczone teoretycznie zgodnie ze wzorem 12:

$$P(k) = \frac{2 * m^2}{k^3} \quad (12)$$

W celu wykazania przybliżonego rozkładu potęgowego $P(k)$ wykreślono na rysunku 3 rozkład potęgowy o współczynniku $\alpha = 3$. Badano sieć bezskalową o parametrach:

- $m_0 = 3$
- $m = 3$
- $t = 3397$



Rysunek 3: Rozkład stopni wierzchołków pojedynczego grafu

Rysunek 3 potwierdza **przybliżony rozkład potęgowy dla wierzchołków**. Stopnie wierzchołków „oscylują” wokół referencyjnego rozkładu. Na rysunkach 4 oraz 5 zobrazowano uśrednione rozkłady $P(k)$ z odpowiednio **10-ciu i stu realizacji** sieci bezskalowej generowanej dla identycznych parametrów.

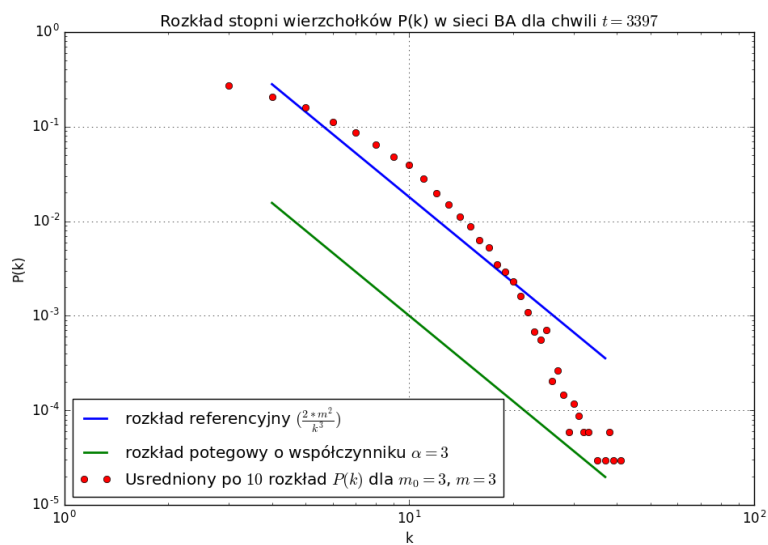
Zbadano również rozkłady dla sieci o parametrach:

Rysunek 6:

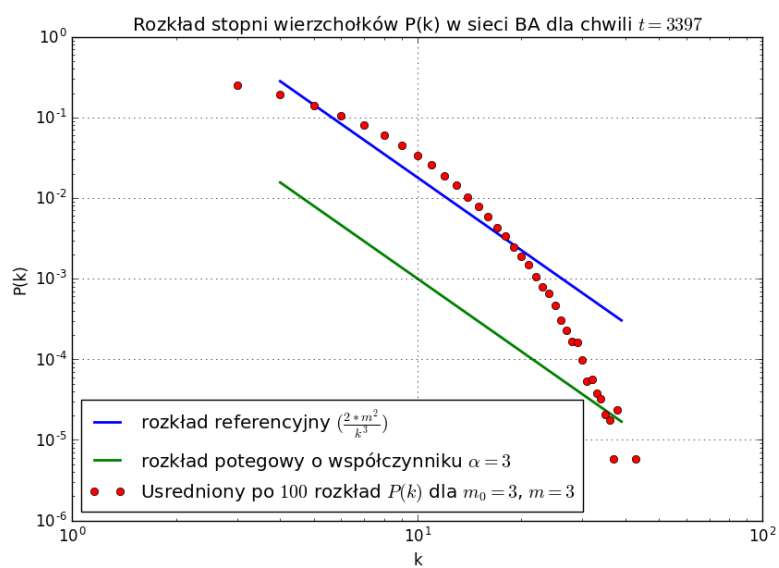
- $m_0 = 5$
- $m = 1$
- $t = 3395$

Rysunek 7:

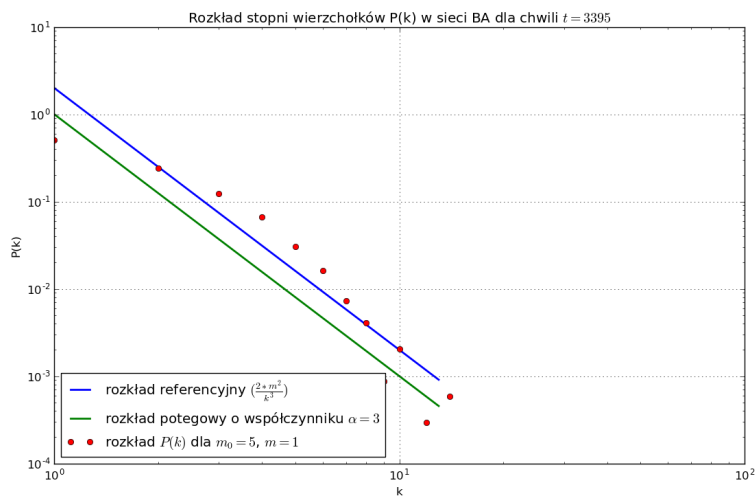
- $m_0 = 2$
- $m = 2$
- $t = 198$



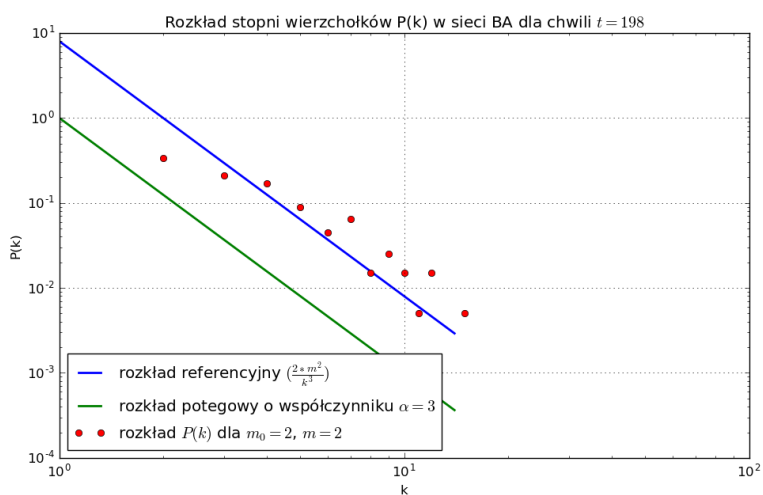
Rysunek 4: Rozkład stopni wierzchołków, uśredniony z 10 prób



Rysunek 5: Rozkład stopni wierzchołków, uśredniony ze 100 prób



Rysunek 6: Rozkład stopni wierzchołków



Rysunek 7: Rozkład stopni wierzchołków

8.2 Propagacja epidemii

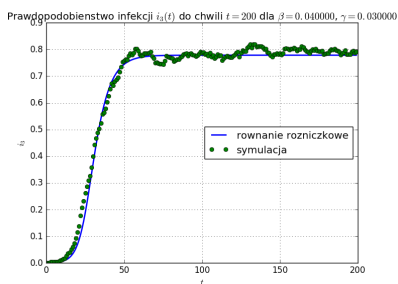
Po wykazaniu poprawności generowanych grafów reprezentujących sieci bezskalowe przystąpiono do realizacji propagacji epidemii. Protokół badań przedstawia się następująco:

- (a) Wygeneruj sieć bezskalową o zadanych parametrach: m , m_0 , t .
- (b) Ustal parametry modelu SIS: β , γ , t_{max} - zakończenie symulacji z tej chwili czasowej, n_{sim} - liczba przeprowadzanych symulacji dla identycznych warunków początkowych (te same zakażone wierzchołki).
- (c) Zakaż wybrany wierzchołek.
- (d) Wylicz wektor $i_k(0)$.
- (e) Wylicz wartość pochodnej dla równania różniczkowego 6. Wylicz wektory $i_k(t)$ dla t dążącego do t_{max} .
- (f) Wylicz rozwiązanie równania różniczkowego dla t zbiegającego do nieskończoności.
- (g) Przeprowadź symulację - wierzchołki „zdrowieją” lub „chorują” z parametrami odpowiednio β , γ .
- (h) Przeprowadź symulację z uśrednianiem.
- (i) Dla zadanego stopnia wierzchołka k wykreśl wykresy przedstawiające zmianę prawdopodobieństwa, że przechowuje on stan „chory” od $t = 0$ do $t = t_{max}$.

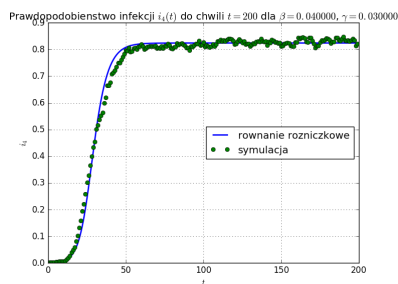
8.3 Symulacja dla pierwszej sieci

Dla każdej próby generowano graf od nowa. Dla sieci bezskalowej o parametrach $m_0 = 3, m = 3, t = 3397$ na poniższym rysunku 8 wykreślono linią niebieską **teoretyczną krzywą dla wektora $i_k(t)$** w kolejnych chwilach czasowych dla wierzchołków w sieci bezskalowej o zadanym stopniu k . Punktami zaznaczono kolejne wartości otrzymane w wyniku symulacji epidemii. Przyjęto następujące wartości parametrów dla epidemii:

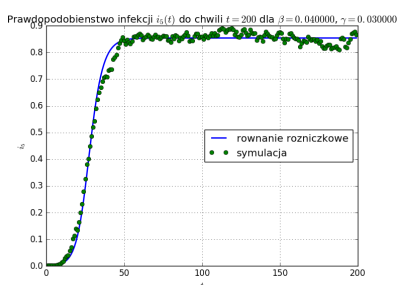
- $k = 3$
- $t_{max} = 200$
- $\beta = 0.04$
- $\gamma = 0.03$
- $\lambda = 1.33$
- teoretyczna wartość progu $\lambda_c = 0.167$
- symulacyjna wartość progu $\lambda_c = 0.174$



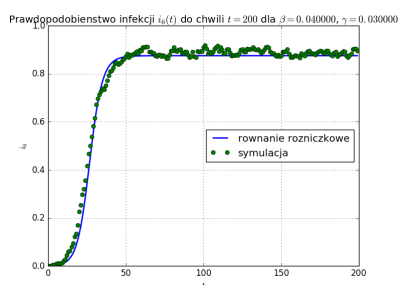
Rysunek 8: Propagacja infekcji dla wierzchołków o stopniu $k = 3$



Rysunek 9: Propagacja infekcji dla wierzchołków o stopniu $k = 4$



Rysunek 10: Propagacja infekcji dla wierzchołków o stopniu $k = 5$

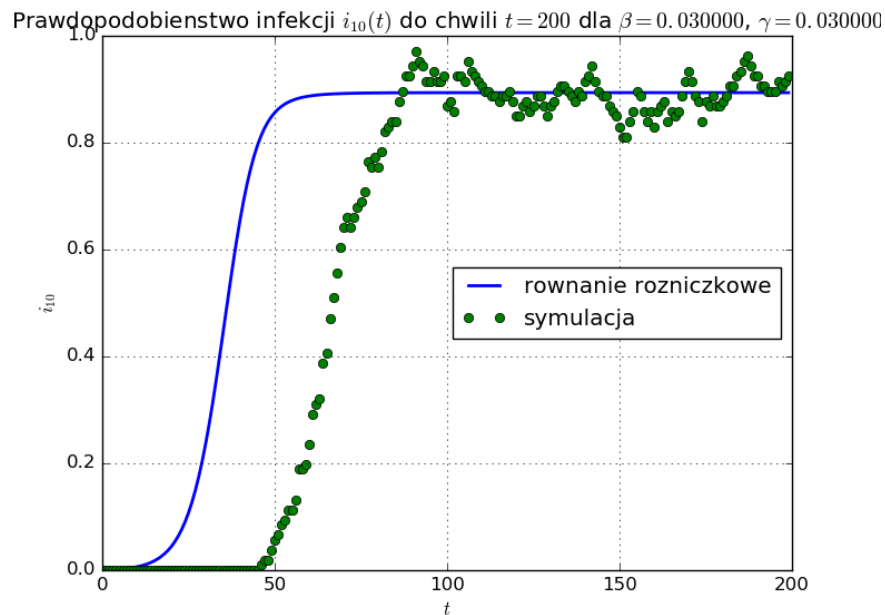


Rysunek 11: Propagacja infekcji dla wierzchołków o stopniu $k = 6$

Na rysunkach 9, 10 oraz 11 wykreślono prawdopodobieństwa infekcji dla wierzchołków o stopniach odpowiednio: $k = 4, 5, 6$.

Następnie przeprowadzono symulację dla tej samej sieci bezskalowej, dla której współczynnik zakażenia jest równy współczynnikowi zdrowienia. Wyniki przedstawiono na rysunku 12.

- $k = 10$
- $t_{max} = 200$
- $\beta = 0.03$
- $\gamma = 0.03$
- $\lambda = 1.0$
- teoretyczna wartość progu $\lambda_c = 0.166$
- symulacyjna wartość progu $\lambda_c = 0.191$

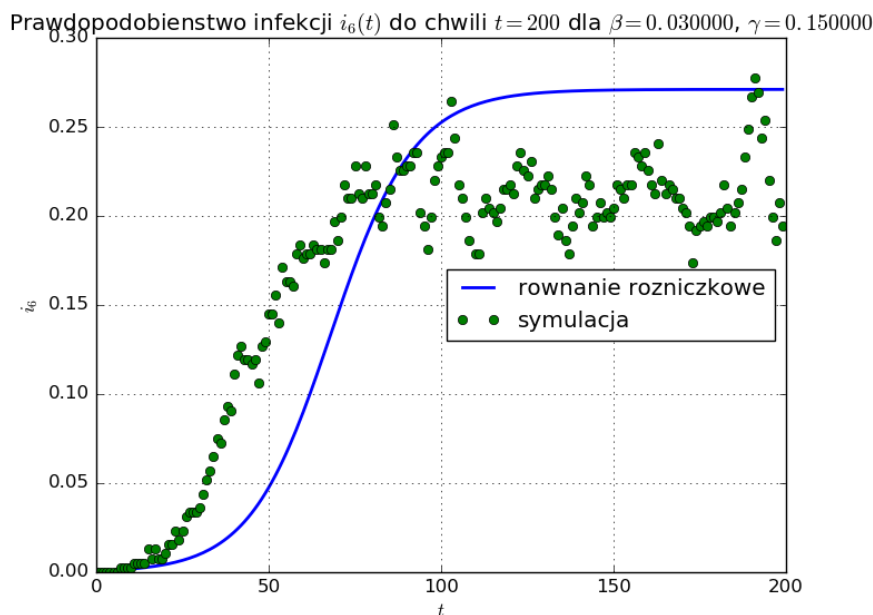


Rysunek 12: Propagacja infekcji dla wierzchołków o stopniu $k = 10$

Symulacja z pewnym opóźnieniem w stosunku do teoretycznego przebiegu ustala się na spodziewanym poziomie.

Kolejnym krokiem było zwiększenie wartości prawdopodobieństwa, że dany osobnik populacji wyzdrowieje (Rysunek 13).

- $k = 6$
- $t_{max} = 200$
- $\beta = 0.03$
- $\gamma = 0.15$
- $\lambda = 0.2$
- teoretyczna wartość progu $\lambda_c = 0.166$
- symulacyjna wartość progu $\lambda_c = 0.178$

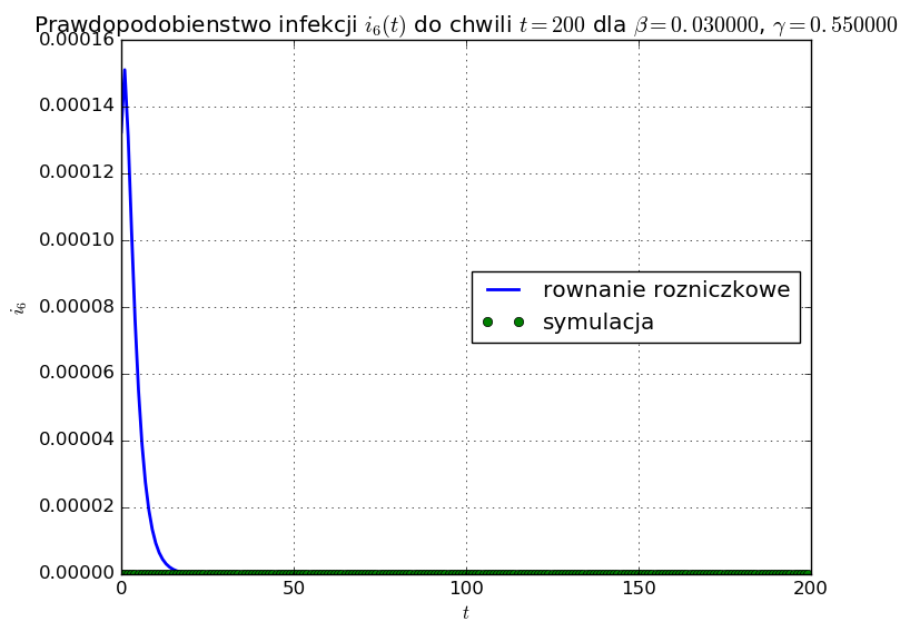


Rysunek 13: Propagacja infekcji dla wierzchołków o stopniu $k=6$

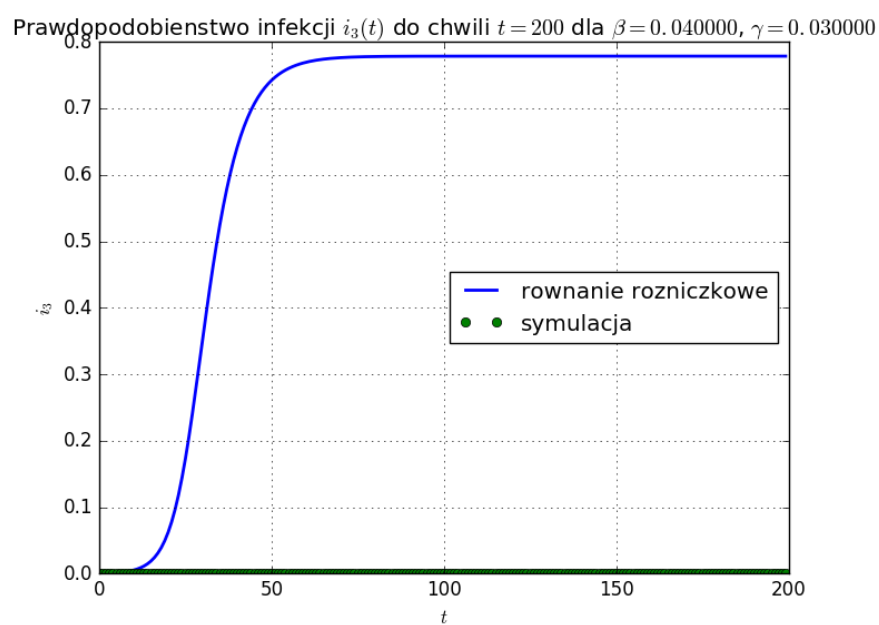
Drastyczne zdrowienie osobników populacji pozwala na stłumienie epidemii (Rysunek 14). Dla tego przykładu wartość $\lambda < \lambda_c$. Przyjęte wartości parametrów:

- $k=6$
- $t_{max}=200$
- $\beta=0.03$
- $\gamma=0.55$
- $\lambda=0.05$
- teoretyczna wartość progu $\lambda_c=0.166$
- symulacyjna wartość progu $\lambda_c=0.190$

Na koniec należy zauważyć, że w niektórych (nielicznych) symulacjach prawdopodobieństwo infekcji było prawie przez całą symulację stałe, równe 0 (rysunek 15). Jest to przypadek szczególny, omówiony zostanie w rozdziale 9.



Rysunek 14: Propagacja infekcji dla wierzchołków o stopniu $k=6$

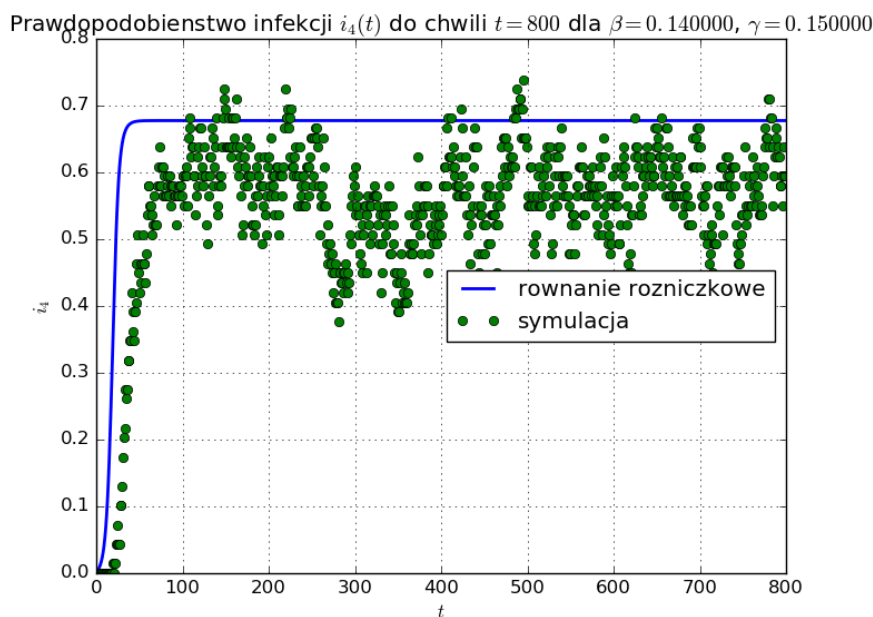


Rysunek 15: Szczególny przypadek wygaśnięcia epidemii

8.4 Symulacja dla drugiej sieci

Następną symulację epidemii przeprowadzono dla sieci bezskalowej o parametrach $m_0 = 3, m = 1, t = 997$. Sieć ta ma strukturę bardzo zbliżoną do drzewa — ale nim nie jest, ponieważ zawiera jedną klikę (początkowy graf). Dla każdej próby generowano graf od nowa. Przyjęto następujące wartości parametrów dla epidemii:

- $k = 4$
- $t_{max} = 800$
- $\beta = 0.14$
- $\gamma = 0.15$
- $\lambda = 0.93$
- teoretyczna wartość progu $\lambda_c = 0.493$
- symulacyjna wartość progu $\lambda_c = 0.499$

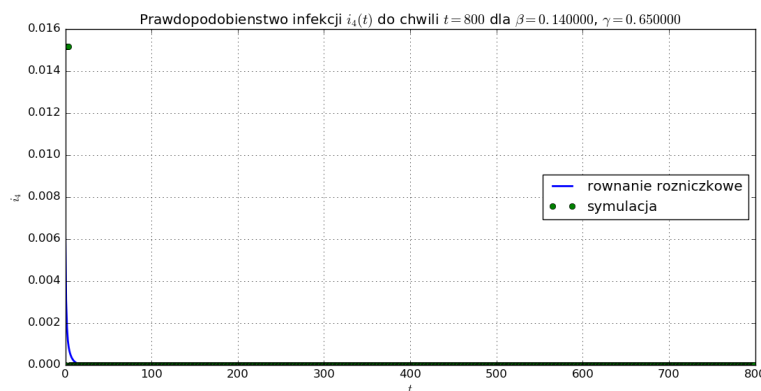


Rysunek 16: Propagacja infekcji dla wierzchołków o stopniu $k = 4$

W czasie niektórych symulacji chorzy zdrowieli wyjątkowo szybko, co powodowało, że epidemia była tłumiona już we wczesnym stadium (jak w przypadku pierwszej sieci, sekcja 8.3). Rysunek ?? ukazuje próbę, w której epidemia rozpropagowała się.

Kolejną symulację przeprowadzono dla sieci bezskalowej o tych samych parametrach ($m_0 = 3, m = 1, t = 997$). Dla tego przypadku wartość $\lambda < \lambda_c$. Przyjęto następujące wartości parametrów dla epidemii:

- $k = 4$
- $t_{max} = 800$
- $\beta = 0.14$
- $\gamma = 0.65$
- $\lambda = 0.21$
- teoretyczna wartość progu $\lambda_c = 0.493$
- symulacyjna wartość progu $\lambda_c = 0.500$



Rysunek 17: Propagacja infekcji dla wierzchołków o stopniu $k = 4$

Początkowe prawdopodobieństwo zakażenia i_k zostało szybko zredukowane. Wyniki pokrywają się z krzywą wyznaczoną teoretycznie.

9 Wnioski

Zaimplementowanie rozwiązania do propagacji epidemii dla populacji modelowanej przez sieci bezskalowe działa poprawnie. Sprawdzono różne konfiguracje sieci bezskalowych oraz zbadano ich rozkłady stopni wierzchołków. Wykazano, że w przybliżeniu realizują one teoretyczny rozkład dany wzorem.

9.1 Współczynniki

Zachowanie sieci zależne jest od współczynników określających prawdopodobieństwo zakażenia się od sąsiada (β) oraz prawdopodobieństwo wyzdrowienia (γ). Ustala się je na początku symulacji. W celu zbadania epidemii przeprowadzono symulacje dla różnych ich wartości. Dowiedziono, że dla sieci bezskalnych trudno jest powstrzymać propagację chorób i dla późnych chwil czasowych ustala się ona na konkretnym poziomie powodując pandemię. Dzieje się tak, ponieważ:

- 1) Teoria (sekcja 3.1: założenia i sekcja 3.3: równanie 11) mówi, że zjawisko to ma miejsce, jeżeli

$$\lambda > \lambda_c$$

gdzie:

$$\lambda = \frac{\beta}{\gamma} \wedge \lambda_c = \frac{\langle k \rangle}{\langle k^2 \rangle}$$

- 2) W sieciach bezskalnych współczynnik λ_c jest bardzo niski, ze względu na ich charakter — a w szczególności występowanie wierzchołków wysokiego stopnia.

Spostrzeżenia dotyczące współczynnika λ :

- 1) Jeżeli $\lambda > \lambda_c$, epidemia rozwinie się i ustali na konkretnym poziomie, przekształcając się w pandemię (np. rysunki 8-11).
- 2) Im współczynnik λ jest wyższy, tym szybciej propaguje epidemia.
 - Dla niskich λ , jednak ciągle wyższych od λ_c , można zaobserwować przesunięcie wykresu wzdłuż osi czasu (OX) w prawo (w kierunku wyższych wartości) oraz zwiększone pochylenie. To oznacza, że choroba wolniej zajmuje nowe wierzchołki na początku (rysunek 13).
- 3) Gdy $\lambda \rightarrow \lambda_c^+$, epidemia wprawdzie dąży do pandemii, jednak dużo mniej stabilnie i zdecydowanie wolniej. Rozwój epidemii jest na początku tłumiony (rysunek 13).
- 4) Jeżeli $\lambda < \lambda_c$, epidemia wygaśnie (rysunek 14),
 - Tym szybciej, im niższy współczynnik λ
 - Zwiększanie współczynnika wyzdrowienia (γ) w stosunku do zachorowania (β) (poprzez działania prewencyjne, profilaktyczne itd.) pozwala stłumić propagację choroby już we wczesnym stadium,

- Należy jednak mieć na uwadze, że stan taki w sieciach bezskalowych występuje dosyć rzadko.

9.2 Symulacja grafowa a rozwiązanie teoretyczne

Różnice w ilości zakażonych w rozwiązaniu teoretycznym i symulacji grafowej wynikają przede wszystkim z tego, że:

- 1) W rozwiązaniu teoretycznym operujemy prawdopodobieństwami i modelujemy rozkład prawdopodobieństwa, niezależny od losowych zmian w grafie,
- 2) W symulacji zliczamy wierzchołki (stosunek liczby zakażonych do całkowitej ich liczby), których stany są zależne od losowych zmian w grafie.

Zdarza się, że epidemia wygasa na samym początku symulacji (rysunek 15). W takiej sytuacji, wynik symulacji będzie całkowicie niezgodny z rozwiązaniem teoretycznym. Dzieje się tak, ponieważ:

- 1) Stosowany jest tutaj losowy algorytm grafowy, każdy wierzchołek zmienia swój stan z określonym prawdopodobieństwem,
- 2) Do zakażenia potrzebny jest kontakt osobnika zakażonego ze zdrowym, czyli musi istnieć osobnik zakażony (musi istnieć wierzchołek w stanie I, mający krawędź do wierzchołka w stanie S),
- 3) Do wyzdrowienia potrzebny jest jedynie osobnik zakażony (wierzchołek w stanie I)
- 4) Jeżeli zdarzy się tak, że pacjent zero wyzdrowieje na samym początku, jedyny wierzchołek o stanie I zmieni stan na S — w sieci pozostaną jedynie wierzchołki w stanie S), epidemia wygaśnie na samym początku i nie będzie miała jak się rozwinąć.

Niezależnie od współczynników, ze względu na losowość zmian stanów wierzchołków i połączeń (np. początek epidemii wśród wierzchołków niskiego stopnia — utrudnione warunki jej rozwoju), epidemia może w początkowej fazie nie rozprzestrzeniać się z dużą prędkością (rysunek 12), ale przyspieszyć kilka momentów t później. Obserwujemy wtedy znaczne przesunięcie fazy gwałtownego wzrostu epidemii dla symulacji grafowej i teoretycznego rozwiązania różniczkowego.

W drugiej badanej sieci można zaobserwować wolniejszą i mniej stabilną propagację (rysunek 16), choć ostatecznie wynik też ustala się na określonym poziomie. Dzieje

się tak, ponieważ druga sieć ma mniej krawędzi (nowe wierzchołki tworzą po jednej krawędzi zamiast trzech — jak w pierwszej sieci), co przekłada się na mniejszą ogólną liczbę połączeń. To z kolei (w rzeczywistości) oznacza mniej rozbudowaną sieć kontaktów między ludźmi, a więc mniejsze możliwości rozprzestrzeniania się choroby.

Bibliografia

- [1] Fronczak Agata, Fronczak Piotr: „*Świat sieci złożonych*”,
Wyd. I, Warszawa, wyd. PWN, 2009, ISBN: 978-83-01-15987-0.
- [2] Fronczak Agata, Fronczak Piotr: „*Epidemie w sieciach złożonych*” [online],
28.10.2010, <http://www.if.pw.edu.pl/~agatka/moodle/epidemie.html> ,
dostęp: 20.11.2016r.