

ANALIZY EPIDEMIOLOGICZNE W ŚRODOWISKU MATLAB/OCTAVE

Agnieszka BARTŁOMIEJCZYK¹, Marcin WATA²

1. Politechnika Gdańska, Wydział Fizyki Technicznej i Matematyki Stosowanej
tel.: 58 347 1347, e-mail: agnbartl@pg.edu.pl
2. Politechnika Gdańska, Centrum Nauczania Matematyki i Kształcenia na Odległość
tel.: 58 348 6195, e-mail: marwata@pg.edu.pl

Streszczenie: W artykule skonstruowano proste modele matematyczne rozprzestrzeniania się chorób zakaźnych oparte na równaniach różniczkowych oraz automatach komórkowych. Na przykładzie modeli SIS i SIR zilustrowano praktyczne zastosowanie pojęć matematycznych nauczanych w toku studiów. Za pomocą symulacji komputerowych, do których użyto pakietów matematycznych MATLAB i Octave, uzyskano wizualizacje tempa rozwoju danej choroby oraz zasięgu epidemii. Ponadto przedstawiono informacje o zastosowaniu e-technologii w kształceniu studentów.

Słowa kluczowe: epidemiologia, modele matematyczne, symulacje numeryczne.

1. INFORMACJE OGÓLNE

1.1. Spojrzenie matematyka na epidemiologię

Epidemiologia jest gałęzią medycyny zajmującą się badaniem przyczyn, rozwoju i szerzenia się chorób zakaźnych, a także innych chorób występujących na dużą skalę w społecznościach ludzkich. Modelowanie rozprzestrzeniania się choroby pozwala na monitorowanie jej przebiegu w określonej populacji oraz daje możliwość ostrzegania przed wybuchem epidemii.

Matematyczne modele epidemiologiczne pomagają wyjaśnić przebieg choroby i pogłębiają zrozumienie wielu procesów związanych z jej rozwojem. W konsekwencji analiza takich modeli pozwala odpowiedzieć na istotne pytania stawiane przez epidemiologów i medyków: czy dojdzie do wybuchu epidemii, jeśli tak, to ile osób zostanie zainfekowanych, a także czy i kiedy choroba zostanie wyeliminowana, [1, 2]. W tym celu wykorzystujemy pojęcie pochodnej, które dobrze charakteryzuje szybkość zmian pewnej wielkości, np. temperatury, ciśnienia czy właśnie rozwoju choroby. Co ważne, pochodna, mówiąc jak szybko rośnie dana wielkość, jednocześnie umożliwia przewidywanie wzrostu liczby zachorowań, np. na grypę czy odrę i ostrzega przed wybuchem epidemii. Użycie pochodnej funkcji jednej zmiennej prowadzi do modeli opisanych równaniami różniczkowymi, których często analityczne rozwiązanie jest zadaniem czasochłonnym bądź też niemożliwym. Wykorzystuje się wówczas technologie komputerowe do przeprowadzenia symulacji numerycznych w zależności od różnych danych początkowych oraz innych parametrów modelu.

1.2. Technologie komputerowe w kształceniu inżynierów

W ocenie i testowaniu proponowanych modeli istotną rolę odgrywa analiza dostępnych danych empirycznych. Umiejętne użycie technik komputerowych pozwala na automatyzację procesu zbierania i analizy tych danych. Ponadto symulacje komputerowe pozwalają określić w przybliżeniu zmianę liczby zainfekowanych i chorych osobników w czasie oraz zasięg epidemii.

Naturalnym wyborem do przeprowadzenia symulacji numerycznych jest MATLAB. Program ten używany jest dość powszechnie przez studentów uczelni technicznych. Początki MATLABA sięgają lat siedemdziesiątych. Od tego czasu jest on intensywnie rozwijany i posiada wiele specjalistycznych pakietów (toolboxów). Pod koniec lat 80. powstał Octave (wolny odpowiednik MATLABA, [3]), który w swoim założeniu, w wielu konstrukcjach oraz poleceniach jest bardzo podobny do MATLABA.

Podkreślimy, że pokazywanie interdyscyplinarności nauki jest bardzo ważne w kształceniu inżynierów na uczelniach technicznych. Przykład modelowania wybuchu epidemii pokazuje konieczność łączenia kilku dziedzin nauki i techniki. Modele biologiczne opisane są w języku matematyki często poprzez równania różniczkowe lub automaty komórkowe. Do efektywnego wyznaczania rozwiązań dla różnych danych początkowych oraz wizualizacji uzyskanych wyników wykorzystujemy techniki komputerowe, np. numeryczne pakiety, ze szczególnym uwzględnieniem pakietu MATLAB. Chcąc połączyć nauczanie tradycyjne przyszłych inżynierów z e-nauczaniem (blended learning) można zamiast MATLABA wybrać Octave w wersji dostępnej online, [4].

2. PROSTE MODELE MATEMATYCZNE

2.1. Model SIS

Przyjmijmy, że w rozważanym modelu liczebność populacji jest stała i równa N , co oznacza, że śmiertelność jest równoważona przez rozrodczość. Zakładamy również, że osobniki, które się rodzą, są zdrowe ale podatne na infekcję. Ponieważ rozważamy populację, która nie nabywa odporności w wyniku infekcji (np. grypy) osobniki dzielimy na dwie grupy, tj.

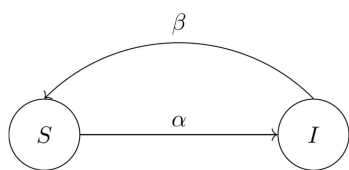
- S – osobniki podatne na infekcję, które nie mają odporności (*susceptible*),

- I – osobniki zainfekowane, które w wyniku kontaktu z innymi mogą przekazywać chorobę (*infected*).

Ponadto przyjmujemy, że choroba rozprzestrzenia się w wyniku kontaktu osobników zainfekowanych i zdrowych, tj. bez udziału pośredniego nosiciela (np. komara w malarii). Liczba takich kontaktów jest proporcjonalna do iloczynu $S(t)I(t)$. Zatem liczba zachorowań jest równa $\alpha S(t)I(t)$. Wówczas rozprzestrzenianie danej choroby reguluje następujący układ równań

$$\begin{aligned}\frac{dS}{dt} &= -\alpha I(t)S(t) + \beta I(t) \\ \frac{dI}{dt} &= \alpha I(t)S(t) - \beta I(t),\end{aligned}$$

gdzie α – średnia liczba kontaktów powodujących infekcję, β – współczynnik wyzdrowień. Ilustrację graficzną przebiegu takiej infekcji widzimy na rysunku 1.

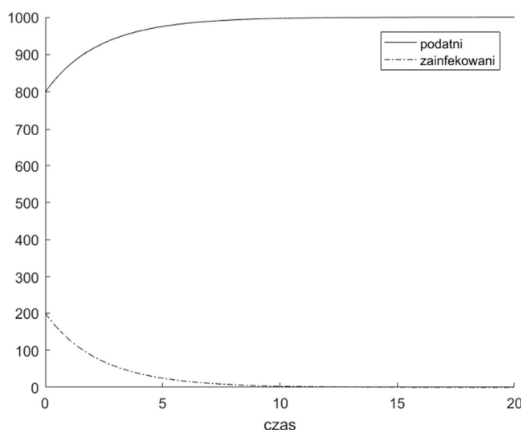


Rys. 1. Schemat przebiegu infekcji typu SIS

Dokonując analizy modelu rozwoju epidemii interesuje nas głównie zmiana liczebności (zagęszczenia) osobników zainfekowanych. O wybuchu epidemii decyduje tzw. bazowy współczynnik odnowienia. Parametr ten oznacza średnią liczbę nowych infekcji (zakażeń) powstałych przez kontakt jednego zainfekowanego osobnika z osobnikami podatnymi i wyraża się następującym wzorem

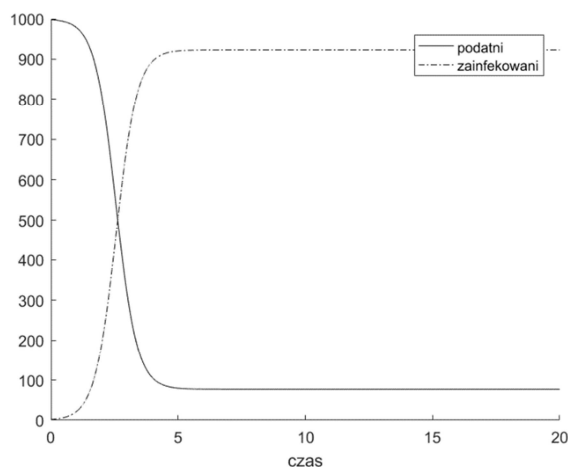
$$R_0 = \frac{\alpha}{\beta} S_0,$$

przy czym S_0 to początkowa liczebność osobników podatnych na daną chorobę. Jeśli bazowy współczynnik odnowienia przyjmuje wartość mniejszą od 1, to choroba wygasa (rysunek 2.), natomiast jeśli jest większy od 1, to może dojść do wybuchu epidemii (rysunek 3.). Zatem jest to jeden z najważniejszych parametrów w analizie epidemiologicznej. Warto podkreślić, że jednym ze sposobów zmniejszenia współczynnika R_0 są szczepienia, w wyniku czego maleje liczba osobników podatnych na infekcję.



Rys. 2. Zanik infekcji w modelu SIS

Na rysunku 3. obserwujemy, że nawet przy bardzo małej liczbie osobników zainfekowanych (dwa osobniki w populacji tysiąca osób) doszło do wybuchu epidemii w bardzo krótkim czasie.



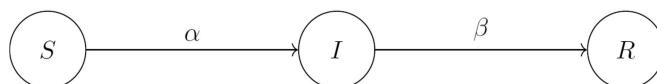
Rys. 3. Wybuch epidemii w modelu SIS

2.2. Model SIR

W celu modelowania przebiegu infekcji z możliwością nabywania odporności (np. jak w przypadku odry) omówimy model SIR (model Kermacka-McKendricka). Populację dzielimy wówczas na trzy grupy, tj.

- S – osobniki podatne na infekcję, które nie mają odporności (*susceptible*),
- I – osobniki zainfekowane, które w wyniku kontaktu z innymi mogą przekazywać chorobę (*infected*),
- R – osobniki, które nabyły odporność na daną infekcję i nie zarażają (*resistant*).

Wówczas rozwój infekcji przebiega według następującego schematu:



Rys. 4. Schemat przebiegu infekcji typu SIR

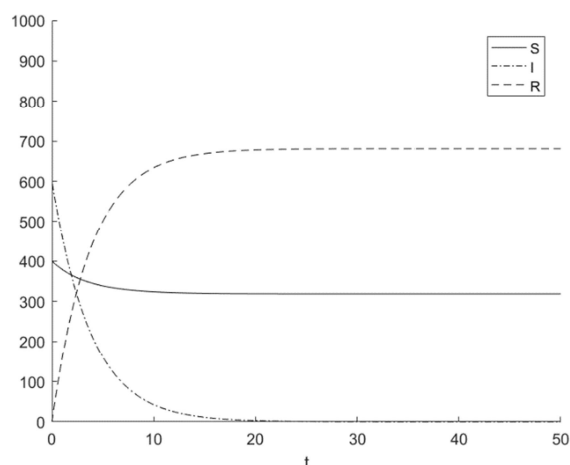
Matematycznie powyższą dynamikę zmian liczby zachorowań w badanej populacji można opisać za pomocą następującego układu równań różniczkowych

$$\begin{aligned}\frac{dS}{dt} &= -\alpha I(t)S(t) \\ \frac{dI}{dt} &= \alpha I(t)S(t) - \beta I(t) \\ \frac{dR}{dt} &= \beta I(t),\end{aligned}$$

gdzie α , jak poprzednio, oznacza średnią liczbę kontaktów powodujących infekcję oraz β jest współczynnikiem wyzdrowień (czyli przejścia osobników zainfekowanych po okresie choroby do grupy osobników odpornych na daną infekcję).

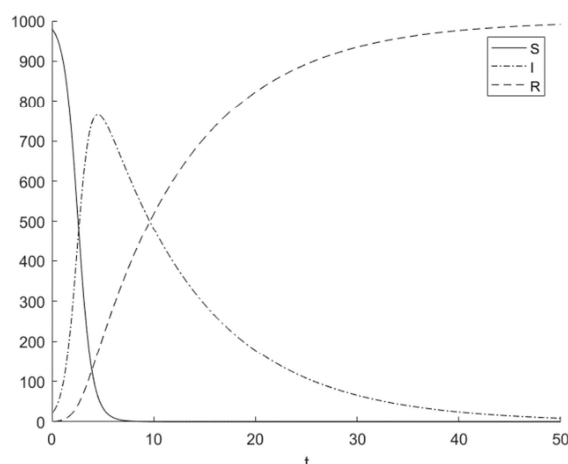
Na poniższych rysunkach przedstawiamy zmianę liczebności poszczególnych grup S , I i R w zależności od wartości bazowego współczynnika odnowienia R_0 . Najpierw rozważamy populację 1000 osobników, w której 600 osobników jest zainfekowanych, a 400 jest podatnych

na infekcję. Zauważmy, że w badanej populacji początkowa liczba osobników odpornych na infekcję wynosi 0, ponieważ założyliśmy, że wszystkie osobniki, które się rodzą są podatne. Podobnie jak w przypadku modelu SIS, pomimo dużej początkowej liczby osobników chorych, epidemia nie wybucha (rysunek 5.).



Rys. 5. Znik infekcji w modelu SIR

Zaobserwujemy, że zwiększając parametr R_0 nawet przy stosunkowo niewielkiej liczbie osobników zainfekowanych dochodzi do wybuchu epidemii. Co więcej, epidemia wybucha w krótkim czasie (rysunek 6.).



Rys. 6. Wybuch epidemii w modelu SIR

Podkreślimy, że w rozważanym modelu SIR, zmienną R interpretujemy jako liczebność osobników odpornych na infekcję. W szczególnej sytuacji R możemy uważać za liczebność osobników usuniętych z populacji (*removed*) z powodu śmierci w wyniku choroby. Prowadzone symulacje mają istotne znaczenie w kontrolowaniu zachorowalności w społeczeństwie. Szybki wzrost liczby chorych np. na odrę może spowodować niemożność zapewnienia opieki wszystkim chorym bądź spowodować spustoszenie jak w przypadku dżumy.

3. SYMULACJE NUMERYCZNE

Symulacje numeryczne zostaną przeprowadzone w celu pokazania możliwości rozprzestrzeniania się chorób

w populacji o ustalonej liczebności i w określonym czasie, przy uwzględnieniu różnych wartości parametrów danego modelu. Podkreślimy, że generowanie takich symulacji stanowi ważny składnik prognozowania medycznego, ale również daje możliwość pokazania studentom, na prostych przykładach, w jaki sposób napisać odpowiedni program oraz interpretować otrzymane wyniki.

W naszej pracy prezentujemy zaimplementowany w MATLABIE kod modelu SIR, który został omówiony w rozdziale 2. Pozostałe implementacje dla modelu epidemiologicznego SIS można wykonać analogicznie. W prezentowanych symulacjach numerycznych użyjemy standardowego solvera `ode45`, który wykorzystuje metodę Rungego-Kutty (ze zmiennym krokiem czasowym dla zwiększenia efektywności obliczeń). Za pomocą wspomnianego polecenia możemy rozwiązywać zagadnienia początkowe postaci

$$\frac{dx}{dt} = f(t, x), \quad x(t_0) = x_0,$$

gdzie t jest zmienną niezależną (interpretowaną jako czas), x jest wektorem szukanych funkcji, $f(t, x)$ jest funkcją t i x , $x = x_0$ jest warunkiem początkowym w czasie t_0 . Składnia polecenia jest następująca:

```
[t,x]=ode45('fname',tspan,xinit, options)
fname – nazwa m-pliku zawierająca zdefiniowaną
w MATLABIE funkcję reprezentującą prawą stronę
równania różniczkowego
tspan – nazwa wektora definiującego przedział
całkowania
xinit – wektor definiujący warunek początkowy
t – wektor wartości zmiennej niezależnej w których
wyliczana jest macierz x
x – jest macierzą o wymiarach length(t) na
length(xinit). Każda kolumna reprezentuje inną
zmienną zależną rozważanego równania.
```

Poniżej przedstawiamy m-pliki MATLABA wykorzystane do graficznej ilustracji dynamiki rozprzestrzeniania się choroby w modelu SIR. Wizualizację zaprezentowaliśmy na rysunkach 5. i 6.

```
fun_sir.m
function dw = fun_sir(t,w)
global alpha beta;

alpha = 0.1*2/1000;
beta = 0.1;

dw = zeros(3,1);
dw(1) = -alpha*w(1)*w(2);
dw(2) = alpha*w(1)*w(2)-beta*w(2);
dw(3) = beta*w(2);
end

sir.m
% Numeryczne rozwiązanie układu równań
różniczkowych w przedziale [0,T]
% dla warunku początkowego [S,I,R]

T = 150; % czas
N = 1000;% populacja
I = 50; %populacja osobników zainfekowanych
S = N-I; % populacja osobników podatnych
R = 0; % populacja osobników odpornych
```

```
[t,y] = ode45('fun_sir',[0, T],[S,I,R]);
```

```
figure
plot(t,y(:,1),'green',t,y(:,2),'red',t,y(
:,3),'blue');
xlabel('t');
legend('S','I','R');
```

Tę samą symulację możemy przeprowadzić w Octave przy użyciu solvera `lsode`. Z uwagi na różnicę w składni polecenia `lsode`, w stosunku do komendy MATLABA `ode45`, w funkcji definiującej układ równań różniczkowych jako pierwszy argument podajemy wektor reprezentujący warunek początkowy, tj. `fun_sir_o(w,t)`. Kolejną różnicą w Octave jest sposób generowania wektora wartości zmiennej niezależnej `t=linspace(0,T,10*T)`, w których wyliczana jest macierz reprezentująca szukane funkcje. Wówczas wywołanie solvera w Octave ma postać `x=lsode('fun_sir_o',[S;I;R],t)`. Wtedy, w wyniku działania skryptu Octave uzyskamy analogiczne wykresy jak na rysunkach 2. i 3. oraz 5. i 6. Jedyne różnice występują w domyślnym formatowaniu osi i legendy.

4. AUTOMATY KOMÓRKOWE

Automaty komórkowe są narzędziem, które jest wykorzystywane do modelowania układów dynamicznych. Pokazują jak proste zasady i lokalne oddziaływania mogą prowadzić do bardzo różnorodnych i skomplikowanych zachowań. Poniżej prezentujemy przykładowy program, napisany w MATLABIE, który ilustruje w jaki sposób automaty komórkowe mogą modelować zjawisko rozprzestrzeniania się epidemii. Wówczas w rozważanym modelu SIR zakładamy, że zarażeniu ulega osobnik mający w swoim otoczeniu przynajmniej jednego chorego.

```
clear all
close all
```

```
% Przestrzeń jest torusem
n = 100; % rozmiar sieci n na n
a = 150; % czas trwania choroby: 1,...,a
T = 400;
p = 0.02; % prawdopodobieństwo
zainfekowania osobnika
g = 0.5; % gęstość zaludnienia
```

```
% Stan początkowy
D1 = ceil(a*rand(n,n)); % macierz
wypełniona liczbami z rozkładu jednostajnego
od 1 do a
D2 = rand(n,n) < p; % wybieramy węzły
zainfekowane
D3 = (rand(n,n) <= g)-1; % wybieramy węzły
niezaludnione
D4 = D3 >= 0; % wybieramy węzły
zaludnione
D = (D1.*D2).*(D4+D3); % plansza łącznie
```

```
%% PROGRAM
t = 1; %czas (np. dni, miesiące)
% tworzymy obraz planszy początkowej w
kolorach
% stan początkowy
green =(D==0) | D<0;
red =(D>0 & D<=a) | D<0;
blue = D>a | D<0;
all_colors = cat(3, red, green, blue);
image(all_colors); title(['t = ',
num2str(t)]) axis square;
```

```
pause(5);
```

```
% początkowa liczba podatnych
S(t) = sum(sum(D==0));

% początkowa liczba zainfekowanych
I(t) = sum(sum((D>0) & (D<=a)));
```

```
% początkowa liczba odpornych
R(t) = sum(sum(D>a));
```

```
up=[2:n 1]; down=[n 1:n-1]; %przestrzeń jest
torusem
```

```
for t = 2:T
D1 = D(:,down); % stan sąsiadów w lewo
D2 = D(:,up); % stan sąsiadów w prawo
D3 = D(down,:); % stan sąsiadów powyżej
D4 = D(up,:); % stan sąsiadów poniżej
```

```
% osobniki mające chorych sąsiadów
sick_neighbors =
((D1>0) & (D1<=a)) + ((D2>0) & (D2<=a)) +
((D3>0) & (D3<=a)) + ((D4>0) & (D4<=a));
```

```
% podatni którzy nie zachorują
stay_healthy =
(D==0) & (sick_neighbors==0);
```

```
% podatni którzy zachorują
get_sick = ((D==0) & (sick_neighbors>0));
```

```
% wszyscy pozostali (w chwili początkowej
są zainfekowani)
others = (D>0);
```

```
% uaktualnianie stanu podatnych
zaczynających chorobę
D(get_sick) = 1;
```

```
% uaktualnianie stanu pozostałych
D(others) = D(others)+1;
```

```
% sumujemy podatnych
S(t) = sum(sum(D==0));
```

```
% sumujemy zainfekowanych
I(t) = sum(sum((D>0) & (D<=a)));
```

```
% sumujemy odpornych
R(t) = sum(sum(D>a));
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% rysujemy stan aktualny
```

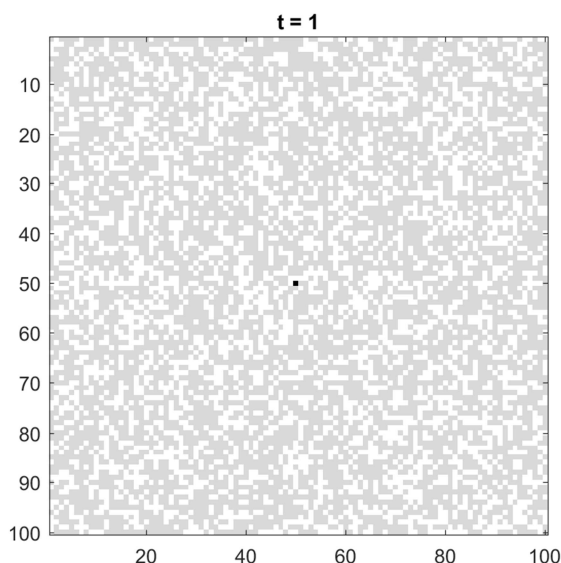
```
green =(D==0) | D<0; %podatni
red =(D>0 & D<=a) | D<0; %zainfekowani
blue = D>a | D<0; %odporni
```

```
all_colors = cat(3, red, green, blue);
image(all_colors); title(['t = ',
num2str(t)])
```

```
axis square;
drawnow
end %koniec pętli for
```

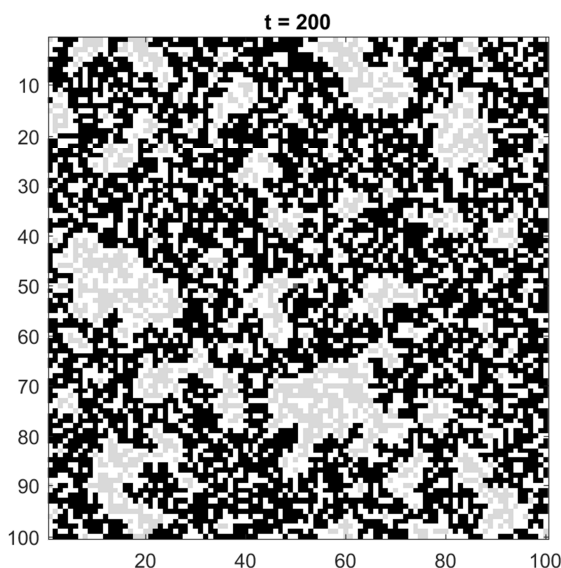
```
figure % S, I i R
t=1:T;
plot(t,S,'green',t,I,'red',t,R,'blue')
legend('podatni','zainfekowani','odporni')
xlabel('czas')
```

Powyższy kod został wykorzystany do wygenerowania rysunków 7-10. Na potrzeby artykułu zamiast kolorów zielonego, czerwonego i niebieskiego użyliśmy skali szarości. Na rysunkach 7-9 każda komórka planszy jest pusta (kolor biały) albo symbolizuje osobnika, który może być w jednym z trzech stanów oznaczonych kolorami (podatny – jasno szary, zainfekowany – czarny lub odporny – ciemno szary). Stan osobnika (komórki) wpływa z określonym prawdopodobieństwem na stan jego bezpośrednich sąsiadów (co najwyżej ośmiu). Jeśli dojdzie do infekcji, to osobnik zainfekowany po czasie trwania choroby staje się odporny. Na rysunku 7. przedstawiamy początkowy stan rozkładu osobników podatnych na infekcję. Osobnika chorego (kolor czarny) umieściliśmy w środku planszy.



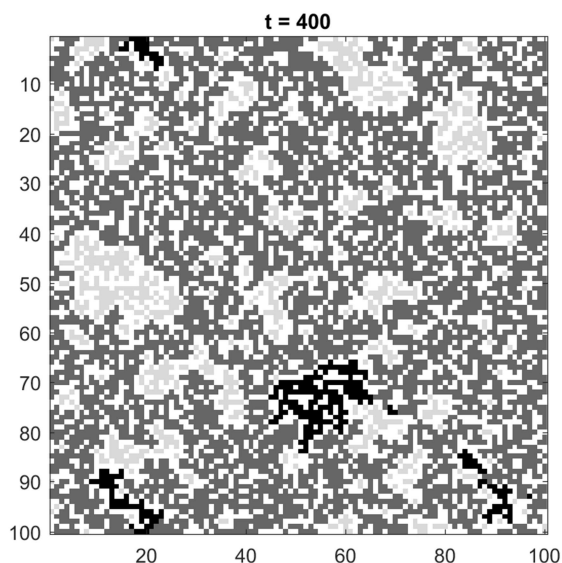
Rys. 7. Początkowy rozkład osobników

Na kolejnych rysunkach obserwujemy dynamikę rozwoju choroby. Rysunek 8. przedstawia rozmieszczenie osobników zainfekowanych po 200 jednostkach czasu (np. dniach, miesiącach) od pierwszego zainfekowania. Po ukończeniu symulacji, czyli po 400 jednostkach czasu (rysunek 9.) widzimy w jaki sposób dana choroba rozprzestrzeniła się.



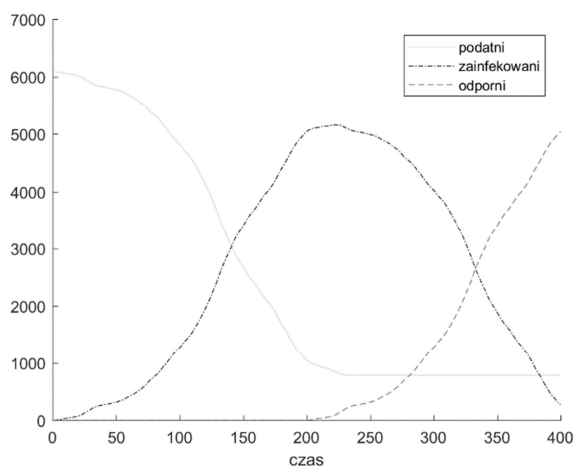
Rys. 8. Rozkład osobników po czasie $t = 200$

Wyniki symulacji, co należy podkreślić, zależą od kilku parametrów. Przede wszystkim od szybkości transferu czynnika chorobowego (np. bakterii, prątka gruźlicy) oraz od początkowego rozmieszczenia osobników (rysunek 7.). Ponadto w zależności od gęstości zaludnienia, choroba może rozprzestrzenić się równomiernie bądź występują obszary nie dotknięte chorobą (jak na rysunku 9.).



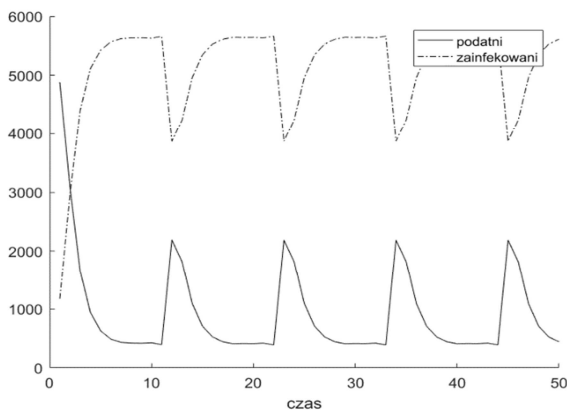
Rys. 9. Rozkład osobników na koniec symulacji

Stosując automaty komórkowe możemy oprócz rozkładu przestrzennego osobników uzyskać dynamikę zmian liczebności poszczególnych grup w modelach SIS i SIR (podobnie jak w modelach opisanych za pomocą równań różniczkowych), rysunki 10. i 11. Ponadto w prosty sposób można uwzględnić czas trwania choroby, co w przypadku stosowania równań różniczkowych prowadzi do równań z opóźnieniem.



Rys. 10. Dynamika zmiany liczebności w modelu SIR

Na rysunku 11. jako konsekwencję stosowania automatów komórkowych obserwujemy cykliczne zmiany liczebności poszczególnych grup w populacji w odróżnieniu od przykładowych rozwiązań otrzymanych na rysunku 3.



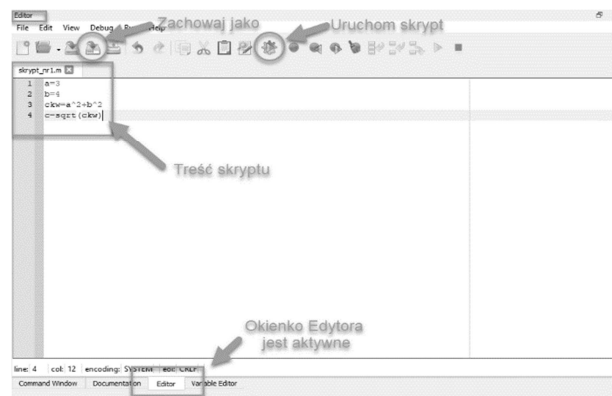
Rys. 11. Dynamika zmiany liczebności w modelu SIS

5. ZALETY BLENDED LEARNING

Podczas zajęć z matematyki dla inżynierów na pierwszym roku studiów omawiane są zagadnienia z wielu działów matematyki (m.in. algebry liniowej, rachunku różniczkowego i całkowego oraz równań różniczkowych). Przyszli inżynierowie powinni zarówno rozumieć wprowadzane w czasie zajęć pojęcia jak i wyrobić sobie biegłość rachunkową. Użycie narzędzi informatycznych może znacząco wspomóc ten proces. Na zajęciach stacjonarnych nie ma zazwyczaj czasu na uczenie studentów posługiwania się takim narzędziem jak MATLAB czy Octave. Rozwiązaniem jest wówczas blended learning, ponieważ naukę MATLABA czy Octave można przygotować w formie e-learningu jako uzupełnienie zajęć stacjonarnych, [5]. Utrudnieniem w przypadku MATLABA jest fakt, że jest to oprogramowanie odpłatne, a tylko niektóre wydziały Politechniki Gdańskiej oferują studentom możliwość uzyskania jego bezpłatnej licencji w wersji online. W przypadku Octave nie mamy takich ograniczeń. Na e-kursie można pokazać jak go zainstalować i uruchomić. Dla studentów, którzy nie mogą zainstalować tego programu na swoim komputerze alternatywą jest posługiwanie się wersją online. Na rysunku 12. pokazany jest zrzut okna Octave z jednego z naszych kursów, demonstrujący jak przygotować i uruchomić prosty skrypt w tym środowisku.

W serwisach oferujących MOOCi znajdziemy kilka interesujących kursów uczących MATLABA i Octave, np.: Coursera oferuje kurs „Introduction to Programming with MATLAB” przygotowany przez Vanderbilt University (USA). Na EdX znajdziemy „Einführung in MATLAB” opracowany przez Technische Universität München, czy też „MATLAB and Octave for Beginners” autorstwa École polytechnique fédérale de Lausanne. Barierą dla studentów może być słaba znajomość języka lub brak czasu na ukończenie dodatkowego kursu. Na naszych e-kursach

przedstawiamy niezbędne minimum informacji o Octave, MATLABIE w celu przeprowadzania obliczeń zarówno w czasie zajęć stacjonarnych jak i w domu. Studenci zdobywają też pierwsze doświadczenia związane ze środowiskiem obliczeniowym, ułatwiające jego wykorzystanie w dalszym toku studiów.



Rys. 12. Zrzut okna Octave

6. WNIOSKI KOŃCOWE

Umiejętność modelowania stanowi nie tylko ważny składnik pracy badawczej, ale również dydaktycznej. Na przykładzie przewidywania wybuchu epidemii pokazaliśmy możliwość zastosowania aparatu matematycznego do konstrukcji wybranych modeli oraz sposób wizualizacji wyników obliczeniowych. Tym samym chcieliśmy podkreślić istotę wspomagania komputerowego zajęć dydaktycznych oraz zapoznania studentów z pracą z danymi i ich wizualizacją w środowiskach MATLAB czy Octave. Zastosowanie metody blended learning umożliwia skupienie się na zajęciach stacjonarnych na interpretacji wyników uzyskiwanych za pomocą skryptów, podczas gdy główny ciężar zrozumienia technicznych aspektów pisania skryptów związany może być z zajęciami przeprowadzonymi online.

7. BIBLIOGRAFIA

1. Foryś U.: Matematyka w biologii, Wydawnictwa Naukowo-Techniczne, Warszawa 2005.
2. Britton N.F.: Essential Mathematical Biology, Springer-Verlag, London 2003.
3. <https://www.gnu.org/software/octave/> (29.06.2019).
4. <https://octave-online.net/> (03.06.2019).
5. Bhatti A., Ranario L.G., Gebreyohannes H.M., Kameswari L.: Using a blended learning approach in teaching mathematics, Proceedings of EDULEARN16 Conference, Barcelona, July 2016, s.1366-1373.

EPIDEMIOLOGICAL ANALYSIS IN MATLAB/OCTAVE

Epidemiology is a branch of medicine that studies the causes, development and spread of infectious diseases, as well as other large-scale diseases in human societies. Information about infectious diseases helps to control the health problem of the society and enables research of the outbreak of an epidemic. In our article, we construct simple mathematical models of the spread of infections: the model of a disease with no immunity against re-infection (SIS model) and the model including such an effect (SIR model). Moreover, we present numerical simulations in MATLAB and Octave, which illustrate the rate and extent of disease epidemics. In addition, information on the use of e-technologies in the education of students is presented.

Keywords: epidemiology, mathematical models, numerical simulations.