

Streszczenie

Wykorzystanie automatów komórkowych do modelowania przebiegu epidemii wybranych chorób zakaźnych

Niniejsza praca ma na celu skonstruowanie programu komputerowego w postaci automatu komórkowego, który symulowałby przebieg i rozwój epidemii chorób zakaźnych w różnych warunkach. Aby spełnić to zadanie, zaimplementowano model matematyczny opisujący zasady transmisji wektorów chorobotwórczych (patogenów) w czasie i przestrzeni z uwzględnieniem czynników społecznych i geograficznych. W pracy kierowano się tym, by w miarę możliwości zaimplementowany model był uniwersalny i umożliwiał użytkownikowi samodzielne dostosowanie parametrów według jego potrzeb. Pozwala to na zobrazowanie zgodne z funkcjonowaniem uproszczonego modelu, jak ewoluowałaby szczegółowo scharakteryzowana choroba w specyficznych warunkach.

Słowa kluczowe: automat komórkowy, epidemia, model epidemiologiczny, patogen

Abstract

Modeling epidemics of selected infectious diseases using cellular automata

A purpose of this work was to build a cellular-automata based computer program to simulate epidemic spread and evolution of infectious diseases under different conditions. To fulfill the task, mathematical model describing rules of time and spatial transmission of pathogenic vector was implemented. The socials and geographical features have been taken into account. The model is constructed to be universal as much as possible and to enable the user to enter his own set of parameters due to his requirements. This allows to show a representation of a simplified model how the detailed characterized disease would evolve in specific conditions.

Key words: cellular automata, epidemy, epidemic model, pathogenic vector

Spis treści

1. Wstęp	11
1.1. Cel pracy	11
1.2. Automat komórkowy (AK)	11
1.3. Choroby zakaźne	16
1.4. Epidemia	17
2. Przegląd literatury	18
2.1. Modele deterministyczne	18
2.2. Modele stochastyczne	20
2.3. Automaty komórkowe	20
2.3.1. Definicja komórki	21
2.3.2. Stosowane modele	21
2.3.3. Ruch komórek	21
2.4. Podsumowanie	22
3. Metodologia	24
3.1. Model epidemiologiczny	24
3.2. Automat komórkowy	27
3.3. Algorytmy	28
3.3.1. Zmiany stanów	28
3.3.2. Ruch osobników	34
4. Implementacja	35
4.1. Menu	35
4.2. Funkcja Run	35
4.3. Funkcja Update	37
4.4. Zmiany stanów	37
4.5. Ruch	38
4.6. Algorytm Floyda-Warshalla	39
5. Wyniki i dyskusja	41

5.1. Losowa dystrybucja populacji	42
5.2. Izolacja zdrowej populacji	45
5.3. Testy statystyczne	48
6. Wnioski	50
Bibliografia	52
Spis rysunków	53
Spis rysunków	54
Spis tabel	54
Spis tabel	55

1. Wstęp

W dobie przeludnienia, gdy roczny przyrost wynosi średnio 76 milionów osób [14], zdrowie publiczne staje się problemem wielkiej wagi. Wysoka koncentracja ludności będąca domeną współczesnego świata stwarza poważne zagrożenie, ze względu na fakt, że gęsto zaludnione aglomeracje są punktami zapalnymi dla szybko rozprzestrzeniającej się fali zachorowań na choroby zakaźne, takie jak grypa, ospa czy gruźlica. Z tego względu niezwykle istotnym jest rozwijanie sposobów przewidywania przebiegu takich zjawisk w zależności od wielu czynników, by móc jak najskuteczniej zastosować odpowiednie środki zapobiegawcze. Zastosowanie automatów komórkowych, które mogą odtwarzać *in silico* to, co obserwujemy w przyrodzie, pomogłoby lepiej zrozumieć, przewidywać i kontrolować potencjalne epidemie.

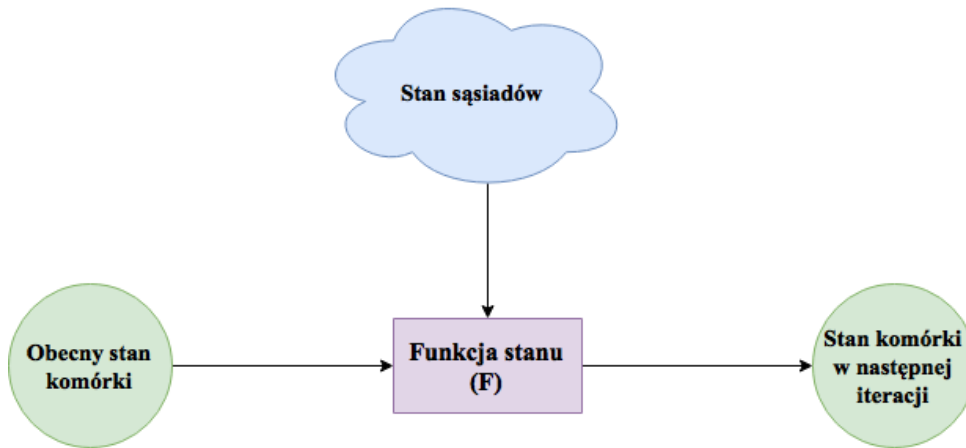
1.1. Cel pracy

W ramach pracy postawiono sobie za cel zgłębienie tematyki modelowania epidemii chorób zakaźnych z wykorzystaniem narzędzi programistycznych w postaci automatów komórkowych. W tym celu dokonano przeglądu istniejących propozycji modeli wysuniętych przez innych twórców, by w oparciu o ich prace, stworzyć własną implementację modelu epidemiologicznego. Dodatkowo skupiono się nad zagadnieniem wpływu sieci komunikacyjnych na transmisję choroby.

1.2. Automat komórkowy (AK)

Twórcą teorii automatów komórkowych był węgierski uczony John von Neumann [15]. W oparciu o maszynę Turinga próbował on stworzyć maszynę zdolną do samopowielania się. Z pośród innych zasłużonych dla rozwijania koncepcji automatów komórkowych można wymienić takich matematyków jak Stanisław Ulam, John Conway czy Stephen Wolfram [8].

Automat komórkowy jest to system złożony z sieci komórek $\{i\}$, mogących przyjąć określony stan z pewnego skończonego zbioru stanów Q . Stany poszczególnych węzłów są uaktualniane w każdej jednostce czasu t na podstawie deterministycznej lub probabilistycznej funkcji stanu (F). Stan każdej komórki w chwili $t + 1$ zależy od stanu tej komórki i stanu komórek w jej sąsiedztwie w chwili t , jak pokazuje schemat na rysunku 1.1.



Rysunek 1.1: Schemat funkcji stanu (F)

$$s_i(t+1) = F(s_j(t)), \quad (1.1)$$

gdzie j należy do $O(i)$, a $O(i)$ jest otoczeniem i -tej komórki [8]

Istnieje wiele metod klasyfikacji automatów. Najprawdopodobniej najpopularniejszą jak dotąd — na podstawie obserwacji ich ewolucji od pewnego przypadkowego stanu wyjściowego — zaproponował jeden z czołowych popularyzatorów AK — Steven Wolfram — w swojej pracy z 1984 r. [18]. Wyszczególnił on cztery klasy automatów:

- automaty jednorodne, gdzie wszystkie węzły systemu przyjmują ten sam stan po pewnym skończonym czasie;
- automaty regularne/periodyczne, gdzie po pewnym skończonym czasie w systemie znajdują się tylko struktury stabilne (tzw. atraktory) lub struktury powtarzalne (tzw. oscylatory);
- automaty chaotyczne, gdzie — zgodnie z teorią chaosu [6] — nawet niewielkie zmiany stanu początkowego układu prowadzą do diametralnie różnych efektów końcowych;

— automaty złożone, które wykazują bardziej złożone zachowanie.

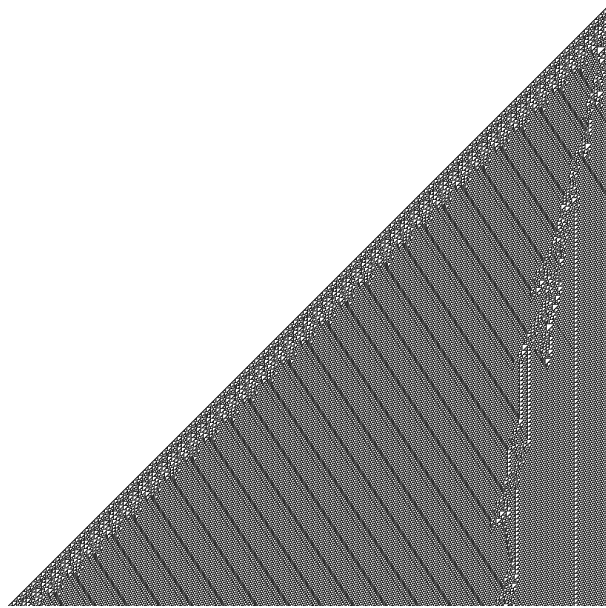
Wyróżnia się wiele automatów komórkowych, różniących się budową w takich aspektach jak:

- wymiar przestrzeni (1D, 2D, 3D, n D);
- liczba stanów komórek (k);
- sąsiedztwo komórek;
- typ komórek;
- warunki brzegowe.

Automat komórkowy w najprostszy sposób można opisać podając dwa parametry - liczbę możliwych stanów komórek (k) oraz promień jej sąsiedztwa (r), czyli maksymalną odległość jaka może dzielić komórki, by miały one wpływ na swój stan. Notacja ta sprawdza się najlepiej w odniesieniu do jednowymiarowych automatów komórkowych, w których komórki położone są względem siebie linearnie i do ich opisu nie jest wymagane określanie typu sąsiedztwa. Najprostszym przykładem automatów komórkowych są elementarne modele jednowymiarowe o dwóch stanach komórki i promieniu sąsiedztwa równym jeden; są one oznaczone zapisem (2,1). Ze względu na to, że każda komórka ma tylko dwóch sąsiadów i może przyjąć jeden z dwóch określonych stanów (0 lub 1), liczba możliwych konfiguracji trzech węzłów wynosi $2^3 = 8$. Ponieważ dla każdej z tych konfiguracji komórka może przyjąć jeden z dwóch stanów, liczba różnych automatów komórkowych o charakterystyce (2,1) wynosi $2^8 = 256$. Elementarne automaty komórkowe przedstawia się poprzez numer reguły w zapisie dziesiętnym, która opisuje wartości, jakie przyjmuje funkcja stanu dla określonych kombinacji komórek w przyjętej kolejności. Przykładowo reguła 110 (ciekawa ze względu na swoją kompletność w znaczeniu Turinga, czyli zdolność do wykonania każdego algorytmu) przedstawiona jest w tabeli 1.1 a jej realizację obrazuje rys. 1.2.

Tabela 1.1: Reguła 110

kombinacja w chwili t	111	110	101	100	011	010	001	000
stan komórki w chwili $t + 1$	0	1	1	0	1	1	1	0

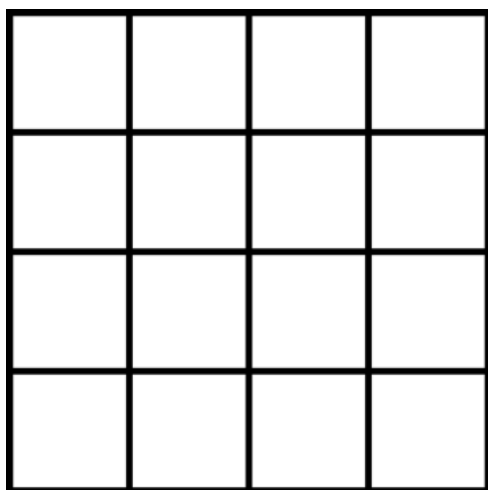


Rysunek 1.2: Przykładowy wynik reguły 110 (źródło: Wikipedia [19])

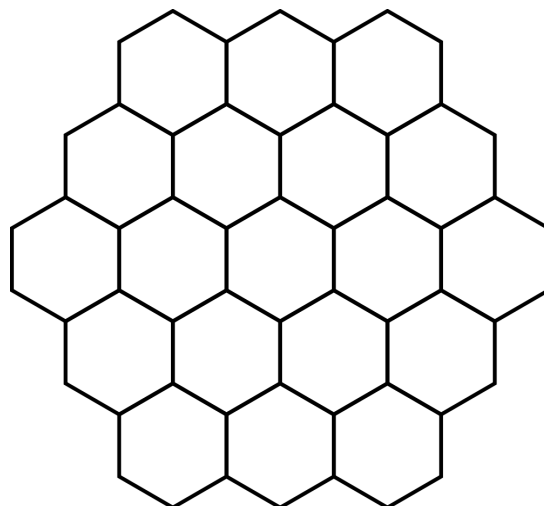
Wśród automatów elementarnych warto wyróżnić dwie podklasy:

- automaty legalne — charakteryzują się zachowaniem symetrii, co oznacza, że wartość funkcji stanu jest taka sama np. dla konfiguracji 100 i 001, a dodatkowo $F(000) = 0$ (tzw. reguła próżni);
- automaty głosujące — biorą pod uwagę tylko liczbę elementów niezerowych w sąsiedztwie (a nie ich wzajemne położenie).

Dwuwymiarowe automaty komórkowe wprowadzają więcej zmiennych, od których zależy przestrzeń automatu. W tym wypadku do scharakteryzowania przestrzeni automatu niezbędne jest określenie typu siatki (rys. 1.3), a także typu sąsiedztwa, z których najpopularniejsze to sąsiedztwo Moore’a i von Neumana (rys. 1.4).



(a)

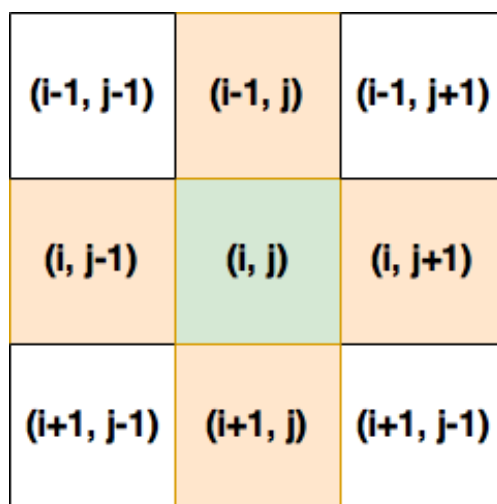


(b)

Rysunek 1.3: Przykładowe rodzaje siatek: a) kwadratowa b) heksagonalna



(a)



(b)

Rysunek 1.4: Typy sąsiedztwa: a) Moore'a b) von Neumana

Bardzo istotną kwestią w przeprowadzaniu symulacji z wykorzystaniem automatów komórkowych jest określenie warunków brzegowych. Wyróżnia się trzy warianty zachowania układu na brzegach siatki:

- warunki periodyczne — komórka położona na brzegu siatki pozostaje w interakcji z komórką znajdującą się na drugim brzegu siatki, tak jakby ze sobą sąsiadowały;
- warunki pochłaniające — sygnał po wykroczeniu poza siatkę całkowicie znika z układu;
- warunki odbijające — komórka na brzegu siatki jest ograniczona przez jej kontur i odbija się od niego.

Automaty komórkowe mają wszechstronne zastosowanie, między innymi jako narzędzia do symulowania układów dynamicznych, co pozwala na odtwarzanie rozmaitych zjawisk fizycznych, chemicznych czy biologicznych.

1.3. Choroby zakaźne

Chorobą zakaźną nazywa się chorobę, będącą skutkiem zakażenia organizmu czynnikiem etiologicznym (zakaźnym) w postaci wirusów, wiroidów, prionów, bakterii lub grzybów. Wbrew powszechnym opiniom za chorobę zakaźną nie uznaje się chorób wywoływanych działaniem pasożytów, jak np. malaria. Celem wirusa (lub innego wektora chorobotwórczego) jest namnożenie się w organizmie swojego gospodarza. Jednakże organizmy żywe wykształciły szereg mechanizmów obronnych związanych z odpornością, tak że jedyną szansą na przetrwanie dla patogenów jest transmisja na inne organizmy. Nosiciel patogenu wykazuje symptomy chorobowe, takie jak kaszel, które pomagają patogenowi przenosić się poza organizm gospodarza i zarażać kolejne osobniki. Zazwyczaj chorobę wywołaną wirusem można podzielić na kilka etapów [10] — okresem utajenia nazywa się czas, kiedy nosiciel patogenu nie jest jeszcze w stanie przenosić go na inne osobniki. Następnie nadchodzi czas, kiedy osobnik zaczyna zarażać inne jednostki, po czym zdrowieje. Okres po zarażeniu, ale przed wystąpieniem objawów nazywa się inkubacją. W tym czasie osobnik może już zarażać inne jednostki, samemu nie będąc świadomym, że istnieje takie zagrożenie (patrz rys. 1.5):



Rysunek 1.5: Etapy infekcji wirusowej wg Scotta i Duncana [10]

W niniejszej pracy rozważano choroby o jednakowym prawdopodobieństwie transmisji w całej populacji, z tego względu wykluczone zostały m.in. choroby przenoszone drogą płciową (*sexually transmitted infections: STI*), takie jak AIDS, na których przekazywanie wpływ mają czynniki związane z aktywnością seksualną i wiekiem jednostek, w związku z czym nie można założyć ich równomiernej dystrybucji.

1.4. Epidemia

Encyklopedia PWN definiuje termin epidemia jako „wystąpienie u ludzi zachorowań na określoną chorobę w określonym czasie i na określonym terenie w liczbie przypadków większej niż przeciętnie” [4]. Oznacza to, że o tym, czy daną liczbę zachorowań można już nazwać epidemią, decyduje charakter samej choroby — tysiące zachorowań na grypę każdego roku nie wzbudzają niczyich obaw, zaś każdy przypadek pojawienia się ospy jest już poważnym zagrożeniem epidemią. W przeszłości epidemie powodowały olbrzymie straty demograficzne i ekonomiczne. Dlatego właśnie tak ważne jest rozwijanie technik umożliwiających szybkie przewidywanie i symulowanie transmisji chorób w określonych warunkach.

2. Przegląd literatury

Na przestrzeni lat rozwinięto wiele różnych podejść do modelowania przebiegu epidemii. Pierwsze próby rozpatrywania tego problemu podjął już w 1766 roku Daniel Bernoulli, bazując na danych pochodzących z epidemii czarnej ospy we Wrocławiu [2]. Modele bazujące początkowo na metodach deterministycznych, zastępowano z czasem podejściem stochastycznym, by wraz z nadejściem ery komputerów wykorzystywać coraz wydajniejsze maszyny do rozwijania technik predykcji w oparciu o programowanie dynamiczne i modele agentowe.

2.1. Modele deterministyczne

Zdecydowanie najpopularniejszym modelem epidemiologicznym, jest model SIR zaproponowany w 1927 roku przez Kermacka i McKendricka [5]. Stworzyli oni model opisywany przez równania różniczkowe, zakładający istnienie pewnej populacji N podzielonej na trzy subpopulacje:

- osobników podatnych (S);
- osobników infekujących (I);
- osobników odpornych (R).

bilansujących się do całkowitej populacji $N = S + I + R$

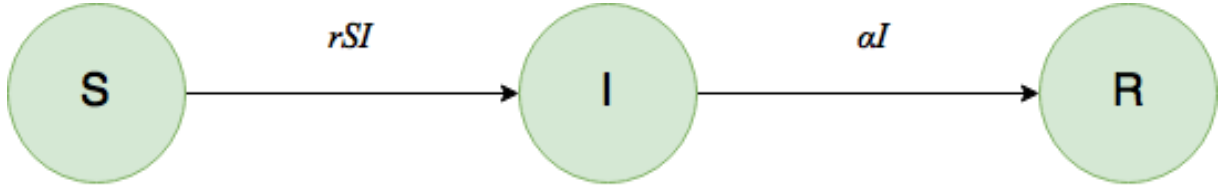
Model opisywany jest przez następujące parametry:

- α — częstość wyzdrowień;
- β — częstość przekazywania patogenu na jeden kontakt;
- C — indywidualne kontakty z innymi osobnikami.

Przy oznaczeniu przez $r = \beta C / N$, model wyrażony jest przez układ równań różniczkowych (2.1):

$$\frac{dS}{dt} = -rSI, \quad \frac{dI}{dt} = rSI - \alpha I, \quad \frac{dR}{dt} = \alpha I \quad (2.1)$$

Zgodnie z zapisem równań różniczkowych (2.1) przepływ z jednej klasy do drugiej odbywa się z zachowaniem ilości osobników. Schemat modelu SIR przedstawia diagram na rysunku 2.1.

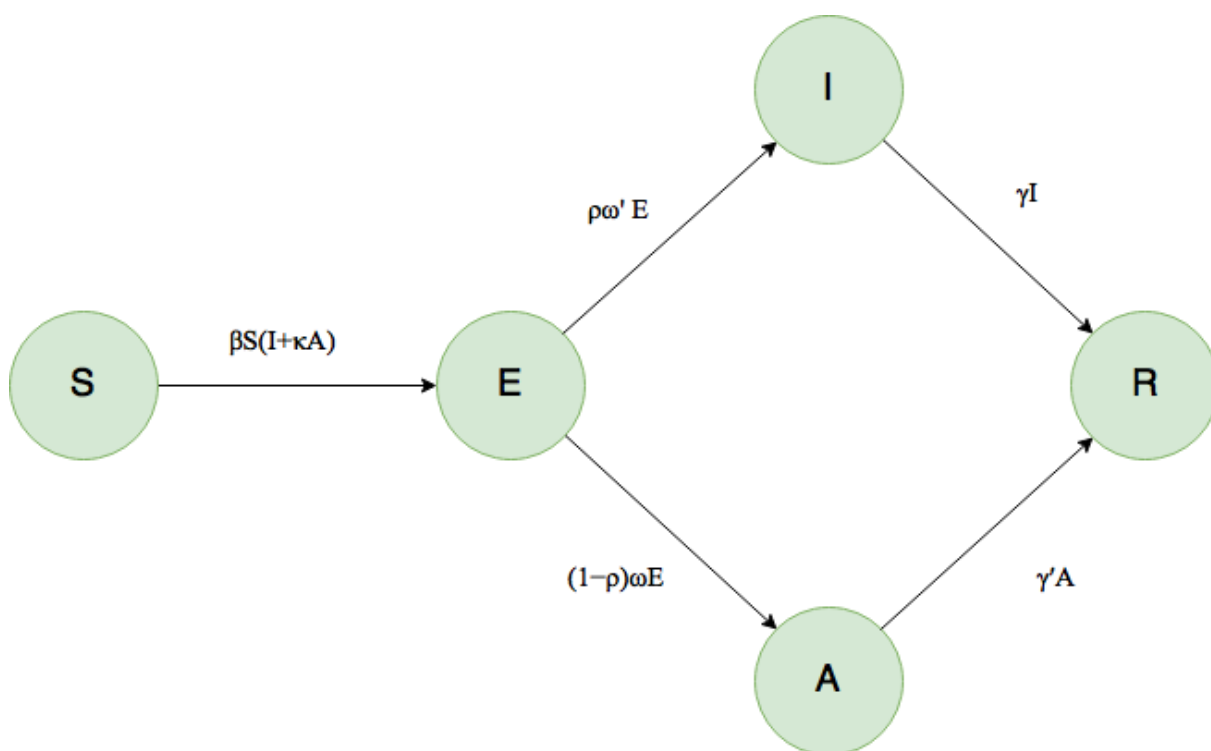


Rysunek 2.1: Schemat modelu SIR

Innymi przykładami modeli deterministycznych są modele SEIR oraz SEIAR. Opierają się na podobnych nieliniowych układach równań różniczkowych opisujących przepływ pomiędzy klasami, wprowadzając jedynie nowe składniki stanu w postaci subpopulacji osób z chorobą w stanie utajonym (E) i subpopulacji gospodarzy patogenów w stanie inkubacji (A). Przedstawiony jako równanie (2.2) układ równań różniczkowych do opisanie modelu SEIAR zaproponowali w swoich badaniach Tianmu Chen i współpracownicy [3]:

$$\begin{cases} \frac{dS}{dt} = -\beta S(I + \kappa A) \\ \frac{dE}{dt} = \beta S(I + \kappa A) - \rho\omega'E - (1 - \rho)\omega E \\ \frac{dI}{dt} = (1 - \rho)\omega E - \gamma I \\ \frac{dA}{dt} = \rho\omega'E - \gamma'A \\ \frac{dR}{dt} = \gamma I + \gamma'A \end{cases} \quad (2.2)$$

Podobnie jak w modelu SIR, także model SEIAR operuje nieliniowymi funkcjami przepływu między populacjami, zachowując całkowitą liczbę osobników. Diagram przejść między stanami zobrazowano na rysunku 2.2.



Rysunek 2.2: Schemat modelu SEIAR

2.2. Modele stochastyczne

Pomimo faktu, że za pomocą deterministycznych metod opartych na równaniach różniczkowych można w sposób dość precyzyjny estymować postępy epidemii w dużych populacjach, Bruno di Stefano i współpracownicy [13] wskazują w swojej pracy, że uproszczenia, które zakładają te metody „(...) w mniejszych [populacjach] mogą prowadzić do nierzetelnych wyników”. Z tego względu modele probabilistyczne wydają się bardziej odpowiednim podejściem do opisywania transmisji chorób zakaźnych.

2.3. Automaty komórkowe

W tej sekcji omówiono i porównano istniejące automaty komórkowe znane z literatury przedmiotu.

2.3.1. Definicja komórki

W zdecydowanej większości prac nad wykorzystaniem automatów komórkowych do modelowania epidemii można spotkać się z pojęciem komórki postrzeganej jako miejsca w przestrzeni, zajmowanego przez pewną populację o całkowitej liczbie osobników N [12], [16], [17], przy czym tylko w jednej z prac S. H. White i współpracownicy zastosowali heterogeniczny rozkład populacji, tak by poszczególne komórki mogły się różnić liczbą przebywających w nich osobników [16]. Na zupełnie inne podejście w kontekście definicji komórki zdecydowali się L. López i współpracownicy [7]. W ich pracy każda komórka systemu reprezentuje indywidualną jednostkę ludzką co, jak przekonują, pozwala na dokładniejsze ustalenie różnych scenariuszy zdarzeń i zastosowanie lepszych strategii kontroli.

2.3.2. Stosowane modele

Niemal wszystkie omawiane w tej sekcji modele bazują na standardowym modelu SIR, opisanym w rozdziale 2.1. Bardzo niewielu autorów decyduje się rozszerzyć podejście o dodatkowe warianty. Wyjątkiem może być praca Ahmeda i Agizy [1], którzy wprowadzili do swojego modelu stan inkubacji i stan utajony choroby. Tak samo postąpił López [7] wysunawszy propozycję modelu SEIAR. Stan utajony (E) to wczesny okres choroby, w którym jednostka jest już zarażona patogenem, ale jeszcze nie zaraża innych osobników. Stan inkubacji (A) obejmuje czas, w którym komórka jest już chora i skutecznie zaraża inne jednostki, ale sama nie wykazuje żadnych objawów choroby. Rozwiązanie polegające na pojawianiu się w populacji nowych jednostek w wyniku narodzin oraz śmierci innych zarówno z przyczyn naturalnych jak i w wyniku choroby zostało zastosowane tylko w modelu wysuniętym przez Lópeza [7]. W przypadku pozostałych prac całkowita liczba populacji pozostaje stała przez cały czas trwania symulacji.

2.3.3. Ruch komórek

Powtarzając za Sirakoulisem [12]: „Jednym z najważniejszych czynników wpływających na rozwój epidemii w populacji jest ruch jednostek”. Istnieją rozmaite podejścia do tego tematu. Ahmed i Agiza [1] całkowicie zrezygnowali z jego uwzględnienia. Część prac [16], [17] wprowadza parametr określający prawdopodobieństwo, że chory

osobnik przetransportuje się z jednej populacji do drugiej i współczynnik połączenia zależny od liczby dostępnych środków transportu między komórkami. Zmienne te są uwzględniane w obliczeniach procentowego udziału chorych w komórce. Innym pomysłem wykazał się López [7], którego model opiera się na pojedynczych osobnikach. W tym przypadku każda jednostka systemu z pewnym prawdopodobieństwem może zmienić miejsce swojego pobytu na losowo wybraną komórkę ze swojego sąsiedztwa bez względu na stan w jakim tamta się znajduje. Autor pracy [7] stwierdza, że można to postrzegać tak, jak gdyby dwie jednostki zamieniły się miejscami.

2.4. Podsumowanie

Omawiane automaty komórkowe należą do klasy IV (automaty złożone), według podziału Wolframa [18]. Krótkie podsumowanie dyskutowanych modeli przedstawia w tabeli 2.1.

Tabela 2.1: Podsumowanie modeli

	White i wsp. [16]	Sirakoulis i wsp. [12]	White i wsp. [17]	López i wsp. [7]	Ahmed i Agiza [1]
Komórka	Populacja	Populacja	Populacja	Jednostka	Populacja
Uwzględnienie inkubacji	NIE	NIE	NIE	TAK	TAK
Uwzględnienie utajenia choroby	NIE	NIE	NIE	TAK	TAK
Uwzględnienie ruchu komórek	TAK	TAK	TAK	TAK	NIE
Uwzględnienie narodzin i śmierci	NIE	NIE	NIE	TAK	NIE

Jak można zauważyć, zdecydowana większość obecnie istniejących automatów komórkowych służących do modelowania przebiegu epidemii bazuje na modelu Kermacka-McKendricka i wykorzystuje metody polegające na wyliczaniu procentowego udziału chorych w populacji reprezentowanej przez poszczególne komórki. Interesującym wydaje się być podejście opisanie każdego węzła systemu jako miejsca bytowania pojedynczego organizmu. Może to przynieść lepsze efekty ze względu na możliwość bardziej szczegółowego scharakteryzowania czynników działających lokalnie. Jedno-

częście takie rozwiązanie pociąga za sobą większe problemy wynikające z zapotrzebowania na moc obliczeniową, jest o wiele bardziej pamięciochłonne i zasobożerne, a odwzorowanie gęstości przestrzennej jest ograniczone do możliwości danych topologią siatki. Dodatkowo większą uwagę należałoby przywiązać do zasad obowiązujących ruch jednostek — sposób polegający na losowym zamienianiu ze sobą miejscami komórek wydaje się mało odpowiadający rzeczywistym zachowaniom ludzkim.

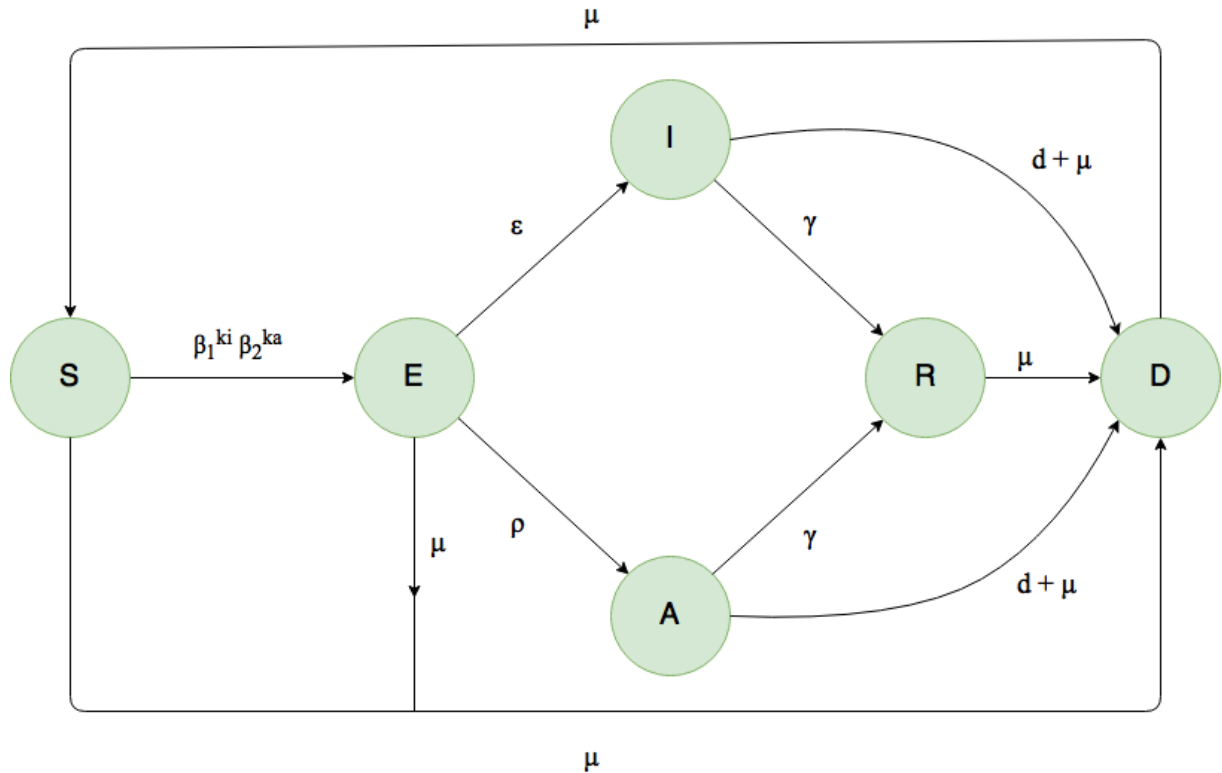
3. Metodologia

Przedstawiony w pracy algorytm oraz realizujący go program komputerowy, jest odtworzeniem pomysłu dotyczącego symulowania przebiegu epidemii w oparciu o model uwzględniający czynniki działające indywidualnie na każdą jednostkę w populacji. Modele epidemiologiczne zawsze obarczone są pewnym błędem wynikającym z konieczności stosowania uproszczonych założeń przy ich wyznaczaniu. Jest to powodowane złożonością i mnogością zmiennych obserwowanych w naturze, które mają wpływ na przebieg procesów zachodzących w naturze. Jak to wykazali M. Shapiro i E. Delgado-Eckert w swojej pracy [11], zarówno wyznaczanie prawdopodobieństwa zachorowania jednostki, jak i próba określenia oczekiwanej liczby chorych w modelach SIR jest problemem klasy NP-trudnej, czyli takiej, których znalezienie rozwiązania nie jest możliwe ze złożonością wielomianową, a sprawdzenie poprawności rozwiązania wymaga co najmniej złożoności obliczeniowej wielomianowej [9]. Z tego względu niniejsza praca nie stara się proponować sztywnych wartości prawdopodobieństw, kładąc bardziej nacisk na dobrze skonstruowany algorytm zależny od zmiennych modyfikowanych przez użytkownika w ramach jego indywidualnych potrzeb.

3.1. Model epidemiologiczny

Do zaproponowanego automatu komórkowego zaimplementowano model typu SE-IAR bazując na modelu zaproponowanym przez L. Lópeza i współpracowników [7]. Model ten jest rozbudowany o dwa dodatkowe stany komórek w stosunku do klasycznych/podstawowych modeli SIR. Wprowadza pojęcie komórki wystawionej na działanie patogenu (*Exposed* - *E*) oraz komórki asymptomatycznej (*Asymptomatic* - *A*), czyli takiej, która mimo wykazywania się zarażliwością wobec komórek w swoim sąsiedztwie, nie przejawia symptomów chorobowych. Takie podejście zapewnia większą swobodę w dopasowaniu modelu do różnych chorób zakaźnych i dlatego zostało obrane na potrzeby programu, który powinien mieć możliwość dostosowania do sze-

rookiej gamy problemów. Dodatkowo model zaproponowany w tej pracy został uzupełniony o możliwość uwzględnienia powrotu komórki ze stanu odporności (R) do stanu podatności (S) po czasie t ustalonym przez użytkownika. Jest to istotne ze względu na istnienie szeregu chorób zakaźnych, po przejściu których nabyta odporność na patogen nie jest odpornością trwałą (jak np. w przypadku zarażenia krztuścem). Ogólny schemat działania modelu przedstawia schemat na rys. 3.1 (por. z rys. 2.1).



Rysunek 3.1: Model SEIAR z prawdopodobieństwami przejść między stanami.

k_i — liczba sąsiadów w stanie *Infected*, k_a — liczba sąsiadów w stanie *Asymptomatic*

- S — osobniki podatne;
- E — nosiciele patogenu w czasie inkubacji;
- I — osobniki chore, przenoszące infekcję;
- A — osobniki chore, przenoszące infekcję, nie wykazujące objawów (asymptomatyczne);
- R — osobniki ozdrowiałe, odporne na działanie patogenu;
- D — pusta komórka (włączając osobniki martwe).

Ponadto wyróżniono dodatkowy stan komórki (P) nie uwzględniany przez model, który reprezentuje przeszkodę w świecie. Jest to komórka niewchodząca w interakcję z innymi komórkami, niemająca żadnego wpływu na stan sąsiadów, jedynie uniemożliwiająca jednostkom zajęcie określonego miejsca na siatce lub jej przekroczenie (jak np. bariery wodne w świecie rzeczywistym). Parametry charakteryzujące dynamikę mogą zostać określone przez użytkownika i są to:

- β_1 — prawdopodobieństwo zarażenia przez komórkę w stanie I ;
- β_2 — prawdopodobieństwo zarażenia przez komórkę w stanie A ;
- ϵ — prawdopodobieństwo przejścia w stan I ;
- ρ — prawdopodobieństwo przejścia w stan A ;
- μ — prawdopodobieństwo naturalnej śmierci i narodzin;
- d — prawdopodobieństwo śmierci w wyniku choroby;
- γ — prawdopodobieństwo powrotu do zdrowia;
- r — promień sąsiedztwa;
- x — prawdopodobieństwo wykonania ruchu przez komórkę;
- y — czas, po którym komórka zatraci odporność na patogen;
- z — prawdopodobieństwo spontanicznego zachorowania.

Parametry x , y i z są opcjonalne i mogą zostać wyłączone z symulacji decyzją użytkownika. W takim przypadku komórki odpowiednio: nie przemieszczają się w trakcie trwania symulacji ($x = 0$), nabywają odporność trwałą po przejściu choroby ($y = 0$), nie mogą zachorować w przypadku braku kontaktu z inną zarażającą jednostką ($z = 0$). Parametr r określa zasięg komórek mogących wchodzić ze sobą w interakcje, tj. maksymalną liczbę komórek, w promieniu której komórka przenosząca chorobę ma szansę na zainfekowanie komórki podatnej. Model korzysta w tym przypadku z oto-

czenia komórek w postaci sąsiedztwa Moore’a. Promień sąsiedztwa będzie się różnił w zależności od tego, czy dana symulacja dotyczy choroby wymagającej bezpośredniego kontaktu do transmisji, czy chorób wywoływanych patogenami posiadającymi możliwość przenoszenia się poza ciałem nosiciela we własnym zakresie (jak np. w przypadku chorób przenoszonych drogą kropelkową). Prawdopodobieństwa przejść pomiędzy poszczególnymi stanami komórek przedstawia tabela 3.1.

Tabela 3.1: Prawdopodobieństwa przejść pomiędzy stanami

Stan	S	E	I	A	R	D
S	$1 - (\mu + \beta_1^{k_i} \beta_2^{k_a})$	$\beta_1^{k_i} \beta_2^{k_a}$	0	0	0	μ
E	0	$1 - (\epsilon + \rho + \mu)$	ϵ	ρ	0	μ
I	0	0	$1 - (\gamma + \mu + d)$	0	γ	$\mu + d$
A	0	0	0	$1 - (\gamma + \mu + d)$	γ	$\mu + d$
R	0	0	0	0	$1 - \mu$	μ
D	μ	0	0	0	0	$1 - \mu$

k_i — liczba sąsiadów w stanie *Infected*, k_a — liczba sąsiadów w stanie *Asymptomatic*

3.2. Automat komórkowy

Prezentowany w tej pracy model symulacyjny jest probabilistycznym automatem komórkowym, co oznacza, że w przeciwieństwie do modeli deterministycznych, funkcja stanu decydująca o zmianie stanów poszczególnych węzłów systemu jest zależna od zmiennych losowych. Dokładny opis funkcji przejścia został opisany w rozdziale 3.1. Symulacja przeprowadzana jest w świecie tworzonym przez dwuwymiarową, kwadratową siatkę. Każda komórka w systemie reprezentuje jednostkę powierzchni, która może znajdować się w jednym z siedmiu możliwych stanów, opisanych przy okazji opisu modelu w rozdziale 3.1. Warunki brzegowe wyznacza kontur zbudowany z komórek w stanie P stanowiąc zamknięcie odbijające i tworzące zamkniętą przestrzeń doświadczalną. Ze względu na charakter symulacji wydaje się to najrozsądniejszym rozwiązaniem w przeciwieństwie do innych możliwości, które wpływałyby na gęstość populacji (zamknięcie pochłaniające) lub stwarzałyby nielogiczne sytuacje, nieopowiadające zachowaniom w ograniczonych obszarach przestrzeni (warunki periodyczne).

3.3. Algorytmy

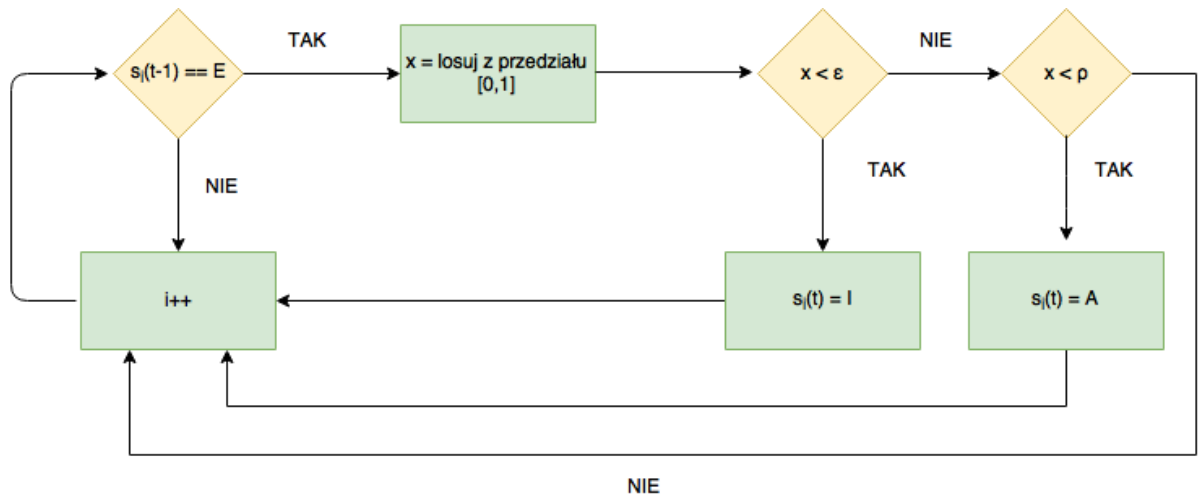
Symulacja rozwoju epidemii przebiega w następujących krokach:

- etap infekcji — nosiciele patogenu przechodzą w stan infekcyjny lub asymptotyczny;
- etap ekspozycji — osobniki podatne są zarażane przez chore jednostki;
- etap spontanicznych zachorowań — osobniki podatne mogą zachorować samorzutnie;
- etap powrotu do zdrowia — osobniki chore zdrowieją i nabywają odporność;
- etap śmierci z powodu choroby — osobniki chore umierają na śmiertelną chorobę;
- etap naturalnych narodzin i śmierci — pojawiają się nowe podatne jednostki i umierają stare;
- ruch komórek — osobniki dążą do obranego przez siebie celu.

3.3.1. Zmiany stanów

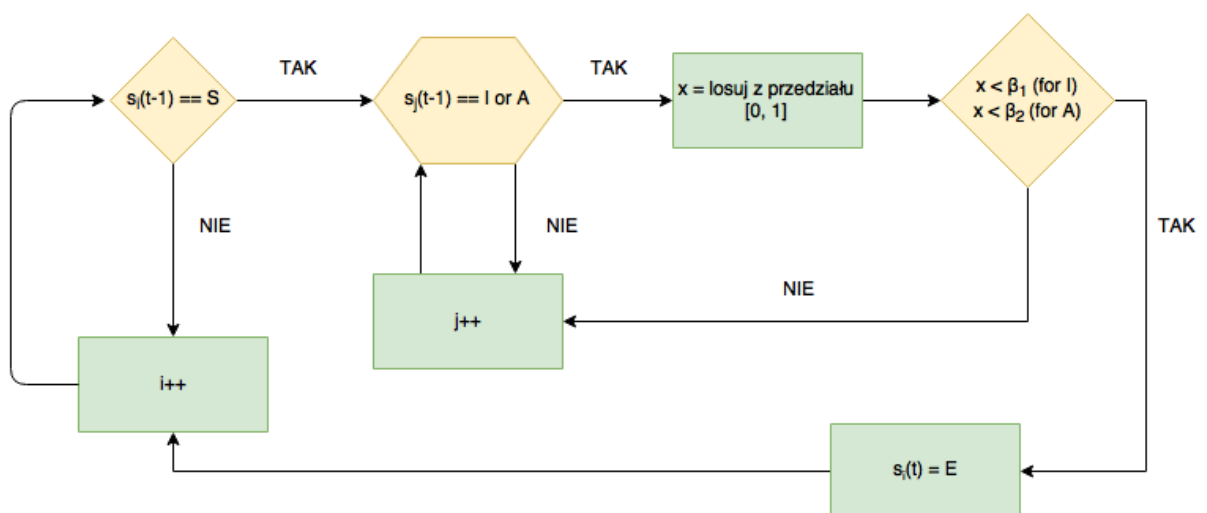
W tej sekcji omówione zostaną szczegółowo zasady, na jakich komórki automatu zmieniają swoje stany.

Infekcja (rys. 3.2) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Jeżeli komórka ta jest nosicielem patogenu, losowana jest pewna liczba z przedziału $[0, 1]$. Liczba ta jest porównywana z prawdopodobieństwem przejścia komórki w stan zainfekowania (I) i, jeżeli to prawdopodobieństwo zostanie spełnione, obecny stan komórki zostaje ustawiony na I . W przeciwnym wypadku sprawdzane jest prawdopodobieństwo przejścia komórki w stan asymptotyczny (A) i jeżeli wylosowana liczba okaże się od niego mniejsza, obecny stan komórki jest zmieniany na A . Jeżeli żadne prawdopodobieństwo nie zostanie spełnione lub komórka miała inny stan niż E przechodzimy do następnego kroku iteracji. Proces ten powtarza dla każdego węzła w systemie.



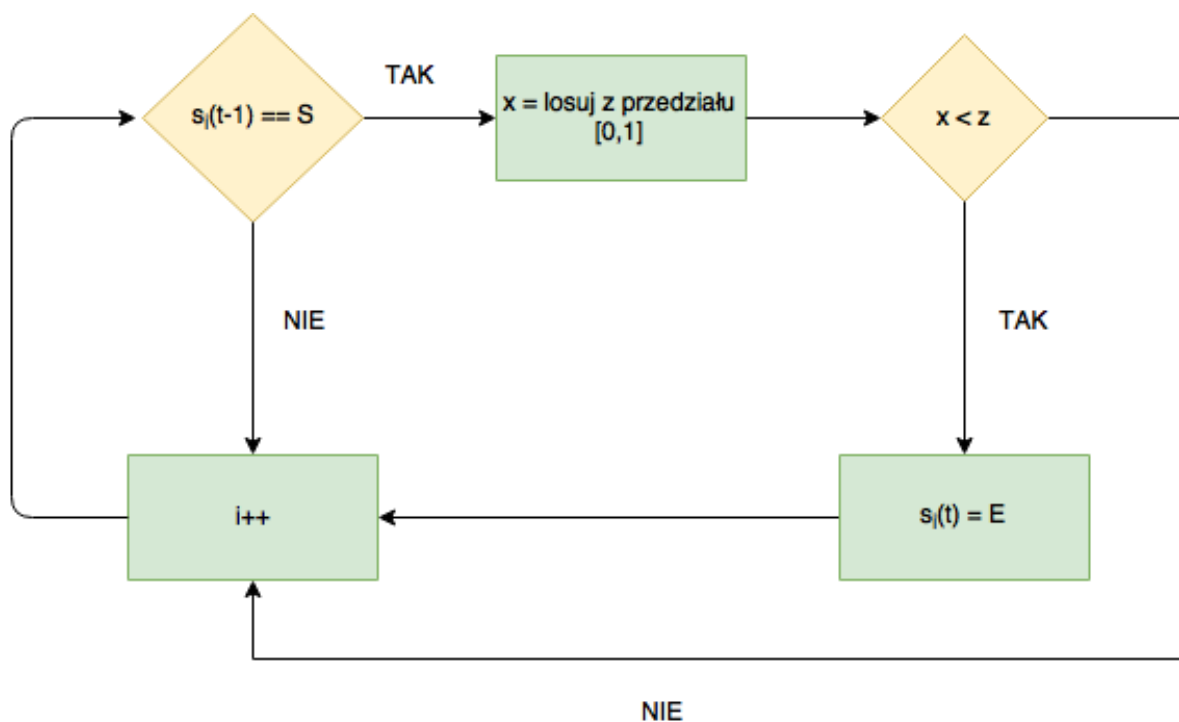
Rysunek 3.2: Algorytm infekcji

Ekspozycja (rys. 3.3) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Jeżeli dana komórka znajduje się w stanie S sprawdzamy stany wszystkich komórek znajdujących się w sąsiedztwie tej komórki określonym promieniem r . Dla każdej komórki o stanie I (*Infected*) lub A (*Asymptomatic*) znajdującej się w zasięgu komórki $s_i(t-1)$, losowana jest pewna liczba z przedziału $[0, 1]$ i porównywana z prawdopodobieństwem przejścia komórki w stan E (β_1 dla komórek w stanie A lub β_2 dla komórek w stanie I). Jeżeli prawdopodobieństwo to zostanie spełnione, komórka podatna zmienia się w nosiciela patogenu. Proces ten powtarza się dla każdego węzła w systemie.



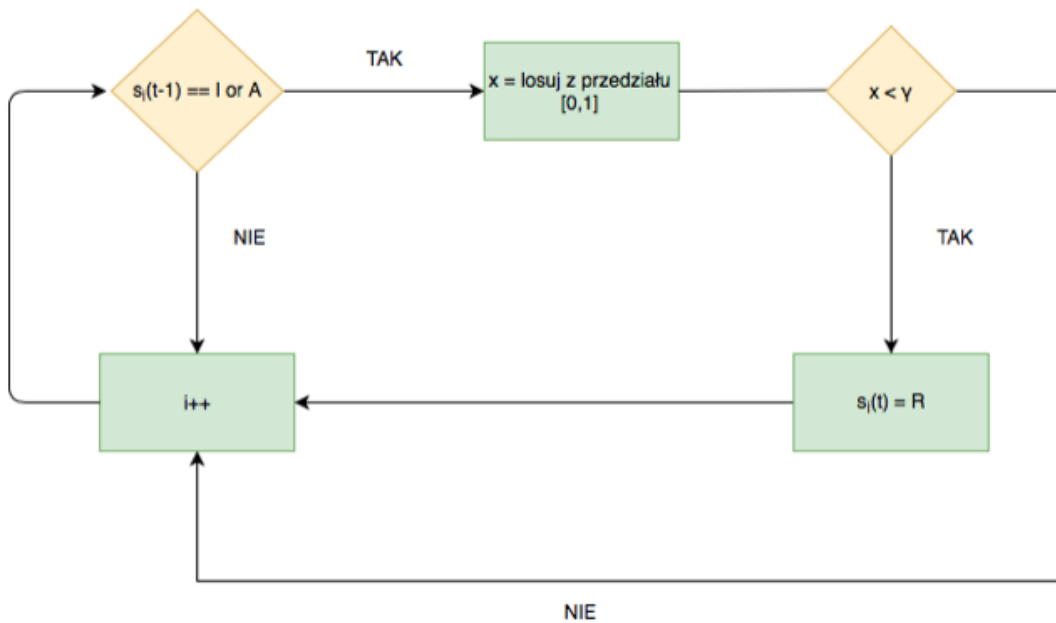
Rysunek 3.3: Algorytm ekspozycji. j należy do $O(i)$, a $O(i)$ jest otoczeniem i -tej komórki [8]

Spontaniczne zachorowania (rys. 3.4) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Jeżeli dana komórka znajduje się w stanie S , losowana jest pewna liczba z przedziału $[0, 1]$. Liczba ta jest porównywana z prawdopodobieństwem spontanicznego zachorowania (parametr z). Jeżeli prawdopodobieństwo to zostanie spełnione, komórka przechodzi w stan E i staje się nosicielem patogenu. Proces ten powtarza się dla każdego węzła w systemie.



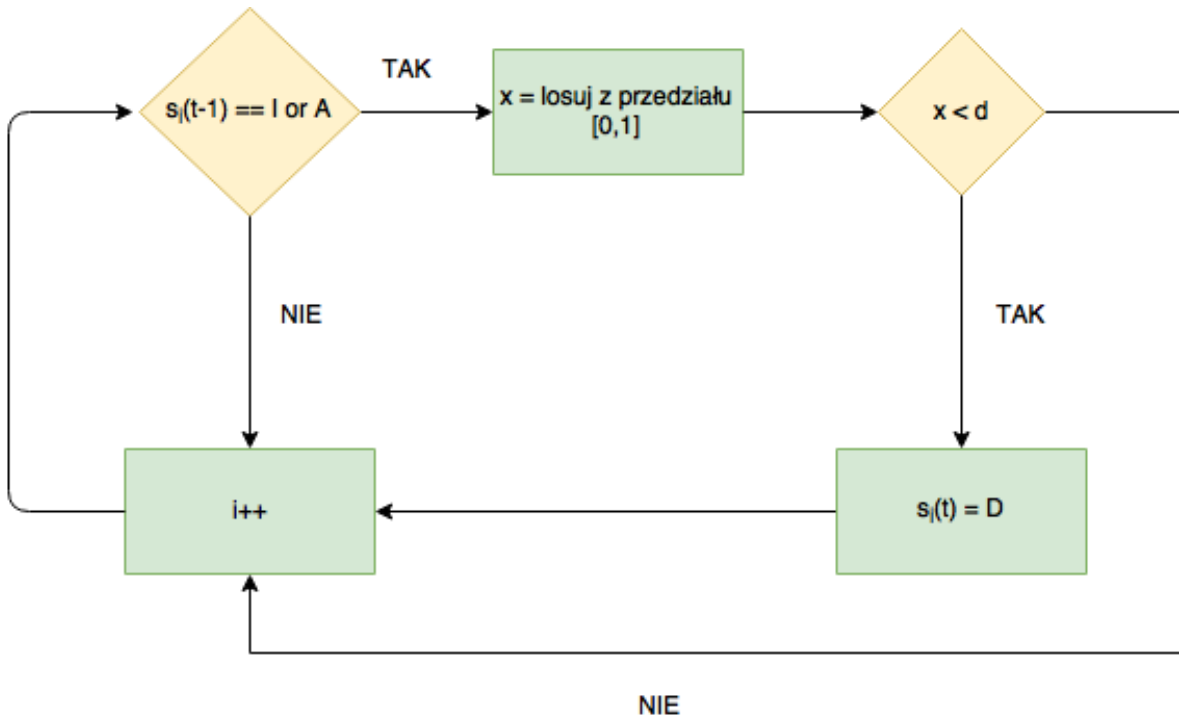
Rysunek 3.4: Algorytm spontanicznych zachorowań

Powrót do zdrowia (rys. 3.5) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Jeżeli dana komórka znajduje się w stanie I lub A , losowana jest pewna liczba z przedziału $[0, 1]$. Liczba ta jest porównywana z prawdopodobieństwem przejścia komórki w stan odporności. Jeżeli prawdopodobieństwo to zostanie spełnione, komórka chora zdrowieje i staje się odporna. W przypadku, kiedy odporność nie jest trwała ($y = 0$) zapisywany jest aktualny czas t w parametrach opisujących klasę komórki, co pozwoli powrócić komórce do stanu podatności po czasie y . Proces ten powtarza się dla każdego węzła w systemie.



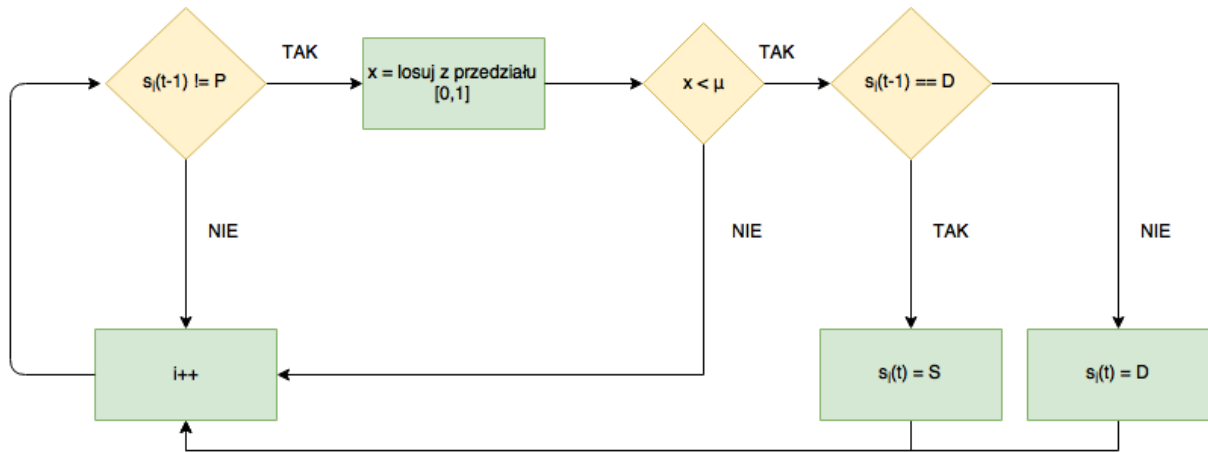
Rysunek 3.5: Algorytm powrotu do zdrowia

Śmierć wywołana chorobą (rys. 3.6) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Jeżeli dana komórka znajduje się w stanie I lub A , losowana jest pewna liczba z przedziału $[0, 1]$. Liczba ta jest porównywana z prawdopodobieństwem, że komórka umrze w wyniku choroby (parametr d). Jeżeli prawdopodobieństwo to zostanie spełnione, komórka chora umiera $s_i(t) = D$.



Rysunek 3.6: Algorytm śmierci wywołanej chorobą

Naturalne narodziny i śmierć (rys. 3.7) — dla każdego węzła w systemie sprawdzany jest jego stan z poprzedniej iteracji. Dla każdej komórki, która nie znajduje się w stanie P , losowana jest pewna liczba z przedziału $[0, 1]$ i jest ona porównywana z prawdopodobieństwem μ . Jeżeli $s_i(t-1) = D$, to $s_i(t) = S$, w przeciwnym wypadku $s_i(t) = D$.



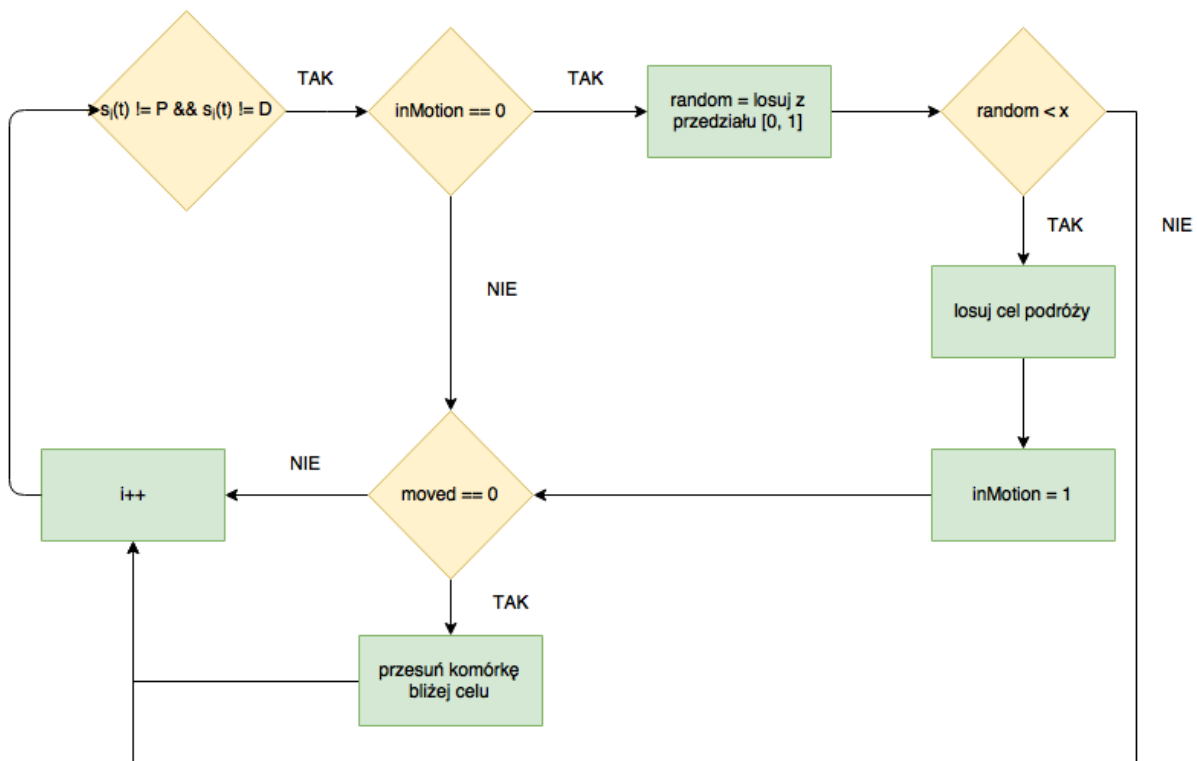
Rysunek 3.7: Algorytm naturalnych narodzin i śmierci

3.3.2. Ruch osobników

Algorytm (rys. 3.8) realizowany jest kilkuetapowo i bierze pod uwagę wartości dwóch zmiennych typu `bool` opisujących własności ruchu każdego węzła w systemie:

- znacznik `inMotion` — dostarcza informację, czy dana jednostka aktualnie jest w trakcie podróży (`inMotion = true`), czy pozostaje w miejscu (`inMotion = 0`);
- znacznik `moved` — informuje system, czy dana jednostka wykonała już swój ruch w obecnej chwili t (`moved = 1`), czy nie (`moved = 0`).

Dla każdego węzła w systemie sprawdzany jest jego stan w obecnej iteracji. Dla każdej komórki, która w danej chwili t się nie porusza (`inMotion = 0`), z wyłączeniem tych w stanie D lub P , losowana jest pewna liczba z przedziału $[0, 1]$ i porównuje się ją z prawdopodobieństwem wykonania ruchu (parametr x). W przypadku spełnienia tego prawdopodobieństwa, komórce wybierany jest w sposób losowy cel jej podróży i oznacza się ją jako komórkę w ruchu (`inMotion = 1`). Następnie każda komórka, która w danej iteracji nie wykonała jeszcze ruchu (`moved = 0`) jest przesuwana w kierunku swojego celu.



Rysunek 3.8: Algorytm ruchu

4. Implementacja

Program został w całości napisany w języku programowania C++, ze względu na jego rozległe zalety, z których najważniejsze to:

- możliwość operacji na referencjach i wskaźnikach;
- dostępność wielu bibliotek;
- jest to język kompilowany, dzięki czemu program w nim działa szybko.

Zastosowane środowiska programistyczne IDE to Dev C++ oraz CodeBlocks.

4.1. Menu

Po uruchomieniu programu następuje początkowa konfiguracja, która obejmuje między innymi ustawienie rozmiaru okna, a także ustawienie punktu startowego dla mechanizmu generowania pseudolosowych liczb całkowitych poprzez wywołanie funkcji `srand()`. Następnie wywoływana jest funkcja sterująca `ShowMenu()`, która wyświetla menu główne programu i zwraca wybór użytkownika powodując wywołanie odpowiedniej funkcji. Użytkownik na tym etapie ma możliwość wyboru poprzez wprowadzenie odpowiedniej opcji z klawiatury:

- S - uruchomienie nowej symulacji
- H - wywołanie funkcji `Help()` zawierającej wskazówki do obsługi programu
- I - wywołanie funkcji `Info()` z krótkim opisem programu
- T - zakończenie działania programu

4.2. Funkcja Run

Po wybraniu opcji Start wywoływana jest funkcja `Run()`, która jest odpowiedzialna za działanie i sterowanie symulacją. Na początku tworzony jest nowy lub nadpisywany istniejący plik `"output.txt"` celem przygotowania dokumentu tekstowego za-

wierającego dane dotyczące aktualnej symulacji. W ramach przygotowania symulacji w funkcji `Run()` wywoływane są kolejno funkcje:

- `GetConditions()` — pobiera od użytkownika szczegółowe dane dotyczące parametrów przygotowywanej symulacji i zapisuje je do zmiennych globalnych, a także zapisuje w katalogu obejmującym plik `"log.txt"` zawierający informacje na temat wybranego zestawu parametrów.
- `SetupArrays()` — jest odpowiedzialna za dynamiczną alokację pamięci dla tablicy:
 - ◇ `cells` będącej siatką komórek tworzących przestrzeń automatu;
 - ◇ `cellsCopy` będącą kopią siatki komórek z poprzedniej iteracji;
 - ◇ `weights` — tablica z zapisanymi wagami przejść pomiędzy węzłami systemu;
 - ◇ `paths` — tablica z zapisanymi najkrótszymi ścieżkami pomiędzy węzłami.
- `SetInitialConditions()` — ustala początkowe wartości zmiennych, zeruje liczniki, ustawia warunki brzegowe, losuje populację na siatce, wstawia komórki w stanie *I*.
- `LoadData()` — funkcja zostaje wywołana pod warunkiem, że opcja ruchu komórek nie została wyłączona. Służy do wczytania do pamięci komputera z zewnętrznego pliku tablicy ścieżek wybranego przez użytkownika wzorca lub, w przypadku braku takiego pliku, stworzenia go samodzielnie.

Po ustawieniu wszystkich parametrów funkcje odpowiedzialne za prowadzenie symulacji i prezentację wyników wywoływane są w pętli `do ... while`, której liczbę powtórzeń określa czas trwania symulacji podany przez użytkownika. Funkcje wywoływane w pętli to:

- `Update()` — uaktualniajaca stany komórek i sterująca ich poruszaniem się;
- `DrawGrid()` — odpowiedzialna za prezentację graficzną układu;
- `ShowLog()` — wyświetlająca parametry charakteryzujące populację (liczby stanów komórek).

4.3. Funkcja Update

Funkcja `Update()` zbudowana jest z dwóch zagnieżdżonych pętli `for` przebiegających po elementach tablicy `cells` z wykorzystaniem zmiennych z zapisaną liczbą wierszy (`rows`) i kolumn (`cols`). Przy każdym wywołaniu kolejne elementy tablicy ulegają kolejno operacjom:

- znacznikowi ruchu (`moved`) każdej komórki jest przypisywana wartość 0;
- wykonywana jest kopia tablicy komórek z poprzedniej iteracji
`(cellsCopy[i][j] = cells [i][j]);`
- instrukcja warunkowa `switch` wywołuje odpowiedni algorytm (z opisanych w rozdziale 3.3.1) w zależności od stanu danej komórki;
- dla każdej komórki wywoływana jest funkcja `NaturalDeathAndBirth()`;
- jeżeli opcja ruchu nie została wyłączona decyzją użytkownika, wywoływana jest funkcja `Move()`, której opis znajdzie się w rozdziale 4.5;
- na koniec stany wszystkich komórek zostają zliczone i zapisane do pliku "output.txt".

Użytkownik w trakcie symulacji ma możliwość jej chwilowego wstrzymania poprzez naciśnięcie spacji lub całkowite przerwanie pętli i powrót do menu, a także wyjście z programu.

4.4. Zmiany stanów

Z poziomu funkcji `Update` wywoływane są funkcje sterujące zmianą stanów komórek. Są to:

- `Infection(i, j);`
- `Exposition(i, j);`
- `SpontaneousInfection(i, j);`
- `Recovery(i, j);`
- `EndResistance(i, j);`
- `DeathByIllness(i, j);`
- `NaturalDeathAndBirth(i, j).`

Funkcje te działają w sposób analogiczny, zatem zostaną opisane zbiorowo. Indywidualne opisy algorytmów można znaleźć w rozdziale 3.3.1. Funkcja jest wywoływana

tylko dla komórek w określonym stanie. Przyjmuje ona na wejściu dwie zmienne typu `integer`, określające pozycję komórki na siatce. Na początku losowana jest liczba z przedziału $[0, 1]$, a następnie jest ona porównywana z prawdopodobieństwem odpowiednim dla danego przejścia. Jeżeli wylosowana liczba będzie mniejsza niż to prawdopodobieństwo, komórka zmieni stan. W przypadku funkcji `Recovery(i, j)` do zmiennej `recDay` opisującej komórkę zapisywany jest stan obecnej iteracji, żeby komórka straciła odporność po upływie czasu zdefiniowanym przez użytkownika (jeżeli odporność nie jest stała).

4.5. Ruch

Do sterowania ruchem komórek służy funkcja `Move()` modyfikująca położenie komórek na siatce w obecnej iteracji. Funkcja przy założeniu, że opcja ruchu nie została wyłączona, wywoływana jest dla każdego węzła w systemie. Najpierw sprawdzany jest znacznik komórki `inMotion`. Jeżeli jest on równy 0, losowana jest pewna liczba z przedziału $[0, 1]$ i porównywana z prawdopodobieństwem wykonania ruchu ustalonym przez użytkownika. Jeżeli prawdopodobieństwo zostanie spełnione, wywoływana jest funkcja `SetDirection()`, która losuje punkt na siatce, do którego dana komórka będzie się zbliżać wraz z inkrementacją zmiennej t . Obranie przez komórkę celu odbywa się z uwzględnieniem zasięgu komórki określonego przez użytkownika podczas ustawiania parametrów symulacji (parametr m). Następnie znacznik `inMotion` jest ustawiany na wartość `true`, informując system, że dana komórka jest w trakcie podróży. Kolejnym etapem jest wykonanie ruchu przez komórkę w kierunku obranego przez nią celu. Każda komórka opisana parametrem `inMotion = true` jest sprawdzana pod kątem tego, czy wykonała już ruch w obecnej iteracji. Jeżeli flaga `mooved` danego węzła przyjmuje wartość `false`, wywoływana jest dla niego funkcja `MoveCell()`. Funkcja ta ma na celu:

- wyznaczenie optymalnego kroku, który zbliży jednostkę do swojego celu;
- przesunięcie jednostki w wyznaczone miejsce.

Obranie kierunku przez komórkę odbywa się z wykorzystaniem algorytmu Floyda-Warshalla (patrz rozdział 4.6) do wyznaczenia najkrótszej ścieżki do miejsca docelowego z uwzględnieniem macierzy odległości pomiędzy poszczególnymi węzłami sys-

temu zaimplementowanej do programu. Kolejnym zadaniem funkcji jest poruszenie jednostki z zachowaniem odpowiednich zasad — jednostka ma prawo przesunąć się na wyznaczone miejsce tylko w dwóch przypadkach:

- jeżeli wyznaczone pole, na które ma się przesunąć, nie jest zajmowane przez żaden inny obiekt (stan = D);
- jeżeli na wyznaczonym do przesunięcia polu znajduje się inna żywa komórka w ruchu i jej następnym krokiem będzie zajęcie pola zajmowanego przez obecnie rozpatrywaną jednostkę. W tym przypadku dochodzi do wyminięcia komórek (zamiany miejscami) i obie jednostki są rozpatrywane jako takie, które wykonały ruch.

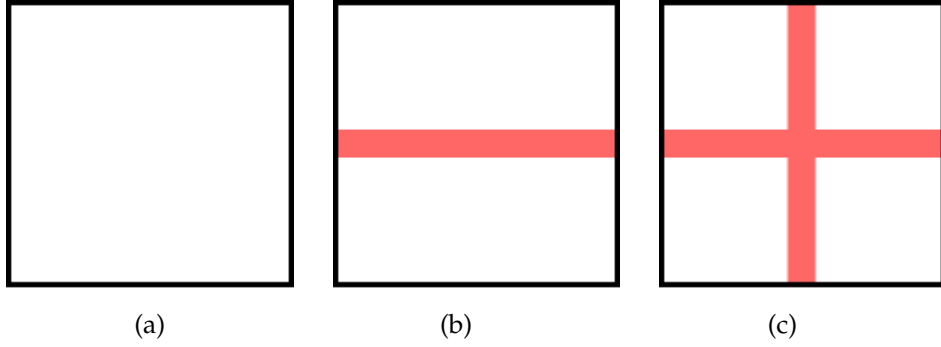
Jeżeli żaden z powyższych warunków nie zostaje spełniony, komórka ma jeszcze prawo do znalezienia obejścia, tak by nie dochodziło do zatorów. W tym wypadku Komórka szuka wolnego pola w swoim najbliższym sąsiedztwie i jeżeli takie znajdzie, ma prawo zająć taką przestrzeń, ale tylko pod warunkiem, że nie oddali jej to od celu.

4.6. Algorytm Floyda-Warshalla

W pracy zwrócono uwagę na wpływ ruchu jednostek na przebieg epidemii. Postawiono pytanie, czy ukierunkowanie ruchu za pomocą ścieżek transportu preferowanych przez jednostki wpłynie na zachowanie systemu. W tym celu w programie zaimplementowano trzy wzorce transportu:

- Wzorzec 1 — z wykorzystaniem macierzy ścieżek bez zróżnicowania wagi przejść pomiędzy komórkami, gdzie koszt ruchu w każdym kierunku jest taki sam (rys. 4.1 a)
- Wzorzec 2 — z wykorzystaniem macierzy ścieżek z prostą trasą szybkiego ruchu, gdzie koszt poruszania się pomiędzy komórkami poza wyznaczoną trasą jest wyższy (rys. 4.1 b)
- Wzorzec 3 — z wykorzystaniem macierzy ścieżek ze skrzyżowaniem szybkiego ruchu, gdzie koszt poruszania się pomiędzy komórkami poza wyznaczoną trasą jest wyższy (rys. 4.1 c)

Do wyznaczenia optymalnej trasy posłużono się algorytmem Floyda-Warshalla, wykorzystującym programowanie dynamiczne do odnalezienia najkrótszych ścieżek mię-



Rysunek 4.1: Wzorce transportu: a) brak preferowanych ścieżek b) preferowana droga w rejonie medialnym siatki c) preferowany układ krzyżujących się tras

dzy wszystkimi parami w grafie. Polega na porównaniu kosztu bezpośredniego przejścia z wierzchołka i do j z kosztem trasy uwzględniającej pośredni wierzchołek k w oparciu o dostarczoną mu macierz W zawierającą koszty przejść pomiędzy wierzchołkami [20]. Algorytm wyrażony matematycznie przedstawia równanie (4.1).

$$W_{i,j}^k = \begin{cases} W_{i,j} & , k = 0; \\ \min(W_{i,j}^{k-1}, W_{i,k}^{k-1} + W_{k,j}^{k-1}) & , k \geq 1. \end{cases} \quad (4.1)$$

5. Wyniki i dyskusja

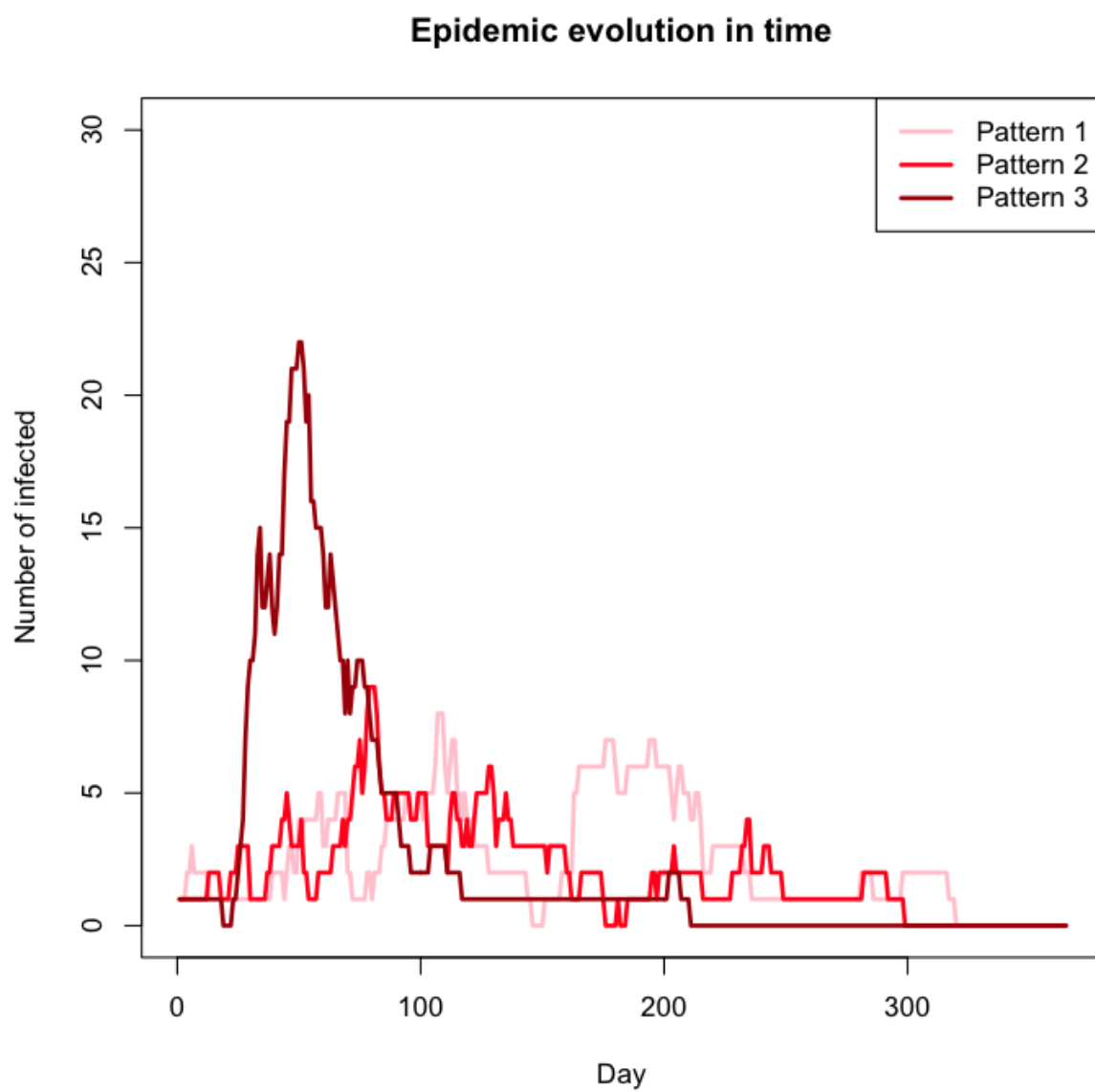
Zaproponowany model pozwala ustalić i sprawdzić różne scenariusze przebiegu epidemii poprzez bardzo szczegółowe ustalenie poszczególnych parametrów mających wpływ na jej rozwój, a zatem umożliwia określenie sposobu w jaki każdy czynnik z osobna wpływa na populację. Szczególną uwagę poświęcono temu, w jaki sposób na rozprzestrzenianie się epidemii wpłynie obecność pewnych preferowanych tras wędrówki jednostek. W celu sprawdzenia zasadności uwzględnienia preferowanych ścieżek obieranych przez osobników w symulowaniu przebiegu epidemii uruchomiono symulację dla trzech zaproponowanych wzorców ścieżek z zastosowaniem tych samych parametrów wejściowych. Parametry użyte w trakcie symulacji przedstawia tabela 5.1.

Tabela 5.1: Parametry symulacji

Parametr	Wartość
Czas trwania symulacji	365
Początkowa gęstość	3%
β_1	0,2
β_2	0,1
ϵ	0,2
ρ	0,3
μ	0,00001
d	0,01
γ	0,05
r	2
x	0,4
y	0
z	0

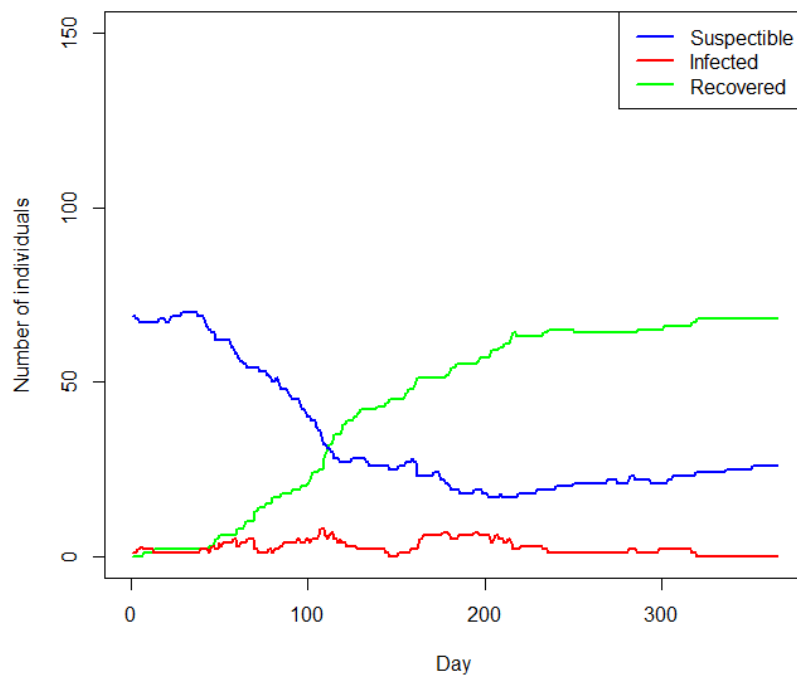
5.1. Losowa dystrybucja populacji

W pierwszym etapie sprawdzono, czy ścieżki komunikacyjne będą miały wpływ na transmisję choroby przy założeniu równomiernego rozkładu populacji na siatce. Wykresy na rysunkach 5.2 i 5.4 przedstawiają przebieg przykładowej symulacji. Jak wynika z wykresów, w przypadku istnienia pojedynczej sieci transportowej epidemia miała bardzo podobny przebieg pod względem czasu rozpoczęcia do całkowitego wygaśnięcia, jak bez zastosowania preferowanych ścieżek, ale liczba zainfekowanych jednostek osiąga wyższe maksimum lokalne. Wyraźnie również widać, że sumaryczna liczba jednostek, które przeszły chorobę, była zdecydowanie wyższa. W przypadku scenariusza zakładającego istnienie dwóch dróg krzyżujących się w centralnej części siatki, dostrzegalne są wyraźne różnice. W tym wypadku najwyższy zapis dotyczący liczby zainfekowanych komórek pojawia się dużo prędzej (około dnia 50. w porównaniu do 90. i 110. w poprzednich scenariuszach), ale również szybciej wygasa. Zarówno maksimum jak i sumaryczna liczba chorych jest znacząco wyższa niż w innych modelach ścieżek. Porównanie liczby zarażonych dla wszystkich wzorców ścieżek przedstawia wykres na rys. 5.1.



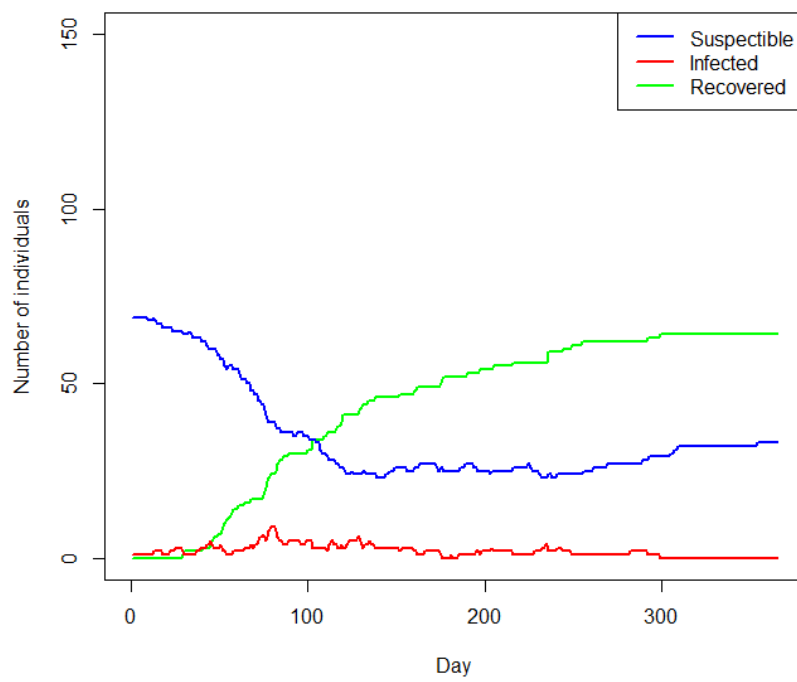
Rysunek 5.1: Liczba zarażonych jednostek

Epidemic evolution in time

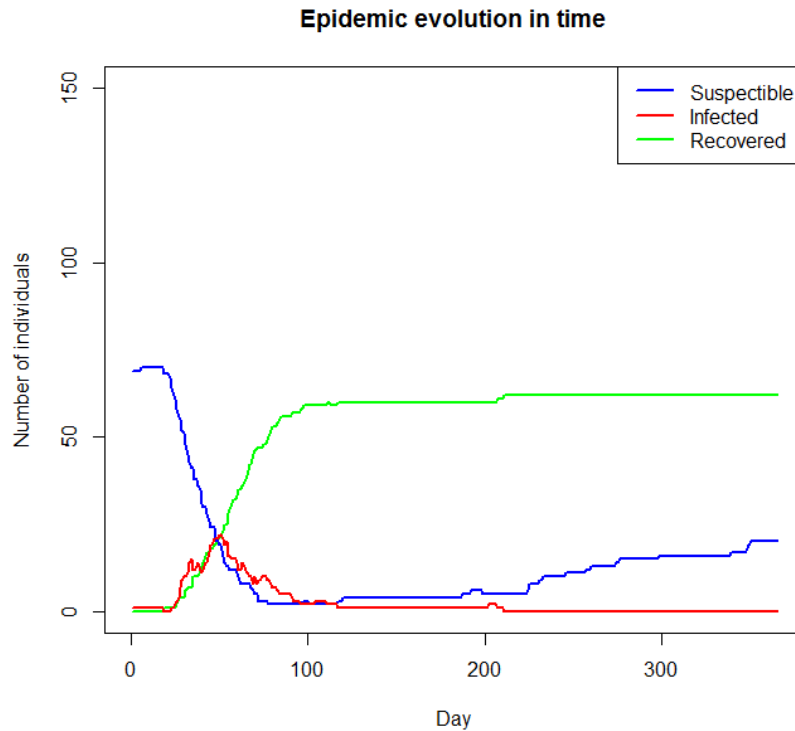


(a)

Epidemic evolution in time



(b)



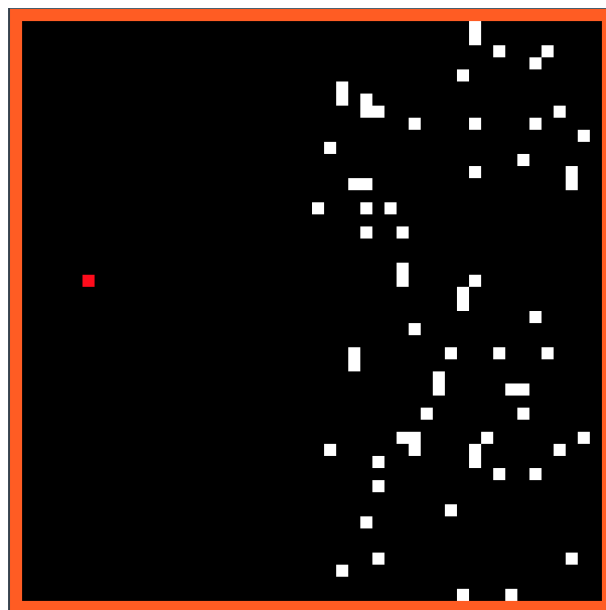
(c)

Rysunek 5.2: Rozwój epidemii w czasie: a) dla wzorca 1 b) dla wzorca 2 c) dla wzorca 3

5.2. Izolacja zdrowej populacji

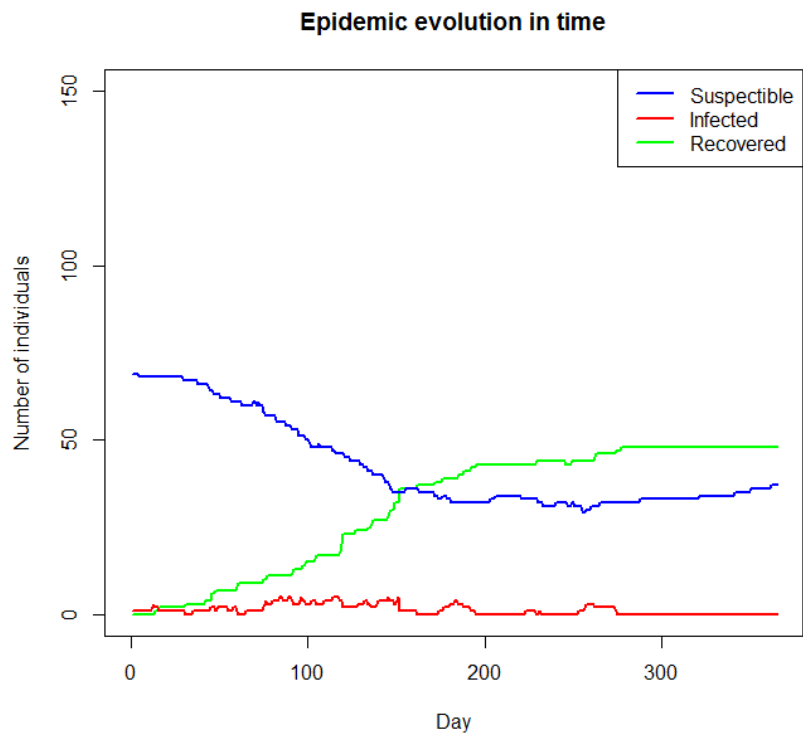
Kolejnym etapem było określenie wpływu obecności często uczęszczanych tras na zarażanie jednostek przez odległe źródło. W tym celu odseparowano zdrową populację poprzez umieszczenie jej na połowie siatki od chorej jednostki, która początkowo zajmowała miejsce na drugiej połowie siatki. Początkowe rozstawienie osobników obrazuje rys. 5.3.

Przy takim założeniu istnienie preferowanych ścieżek ma jeszcze większe znaczenie dla rozwoju epidemii. Wielokrotnie powtarzane symulacje wykazały, że w przypadku braku często obieranej wspólnej drogi (wzorzec 1) epidemia rozwija się średnio 1 raz na 10 prób. Jej przebieg zaś w porównaniu do symulacji przeprowadzonych na innych wzorcach ścieżek jest podobny do tego, w których jednostki nie były od siebie początkowo izolowane, tj. trwa najdłużej i charakteryzuje się niższą szczytową wartością zarażonych jednostek oraz sumaryczną liczbą osobników, które przeszły chorobę. W

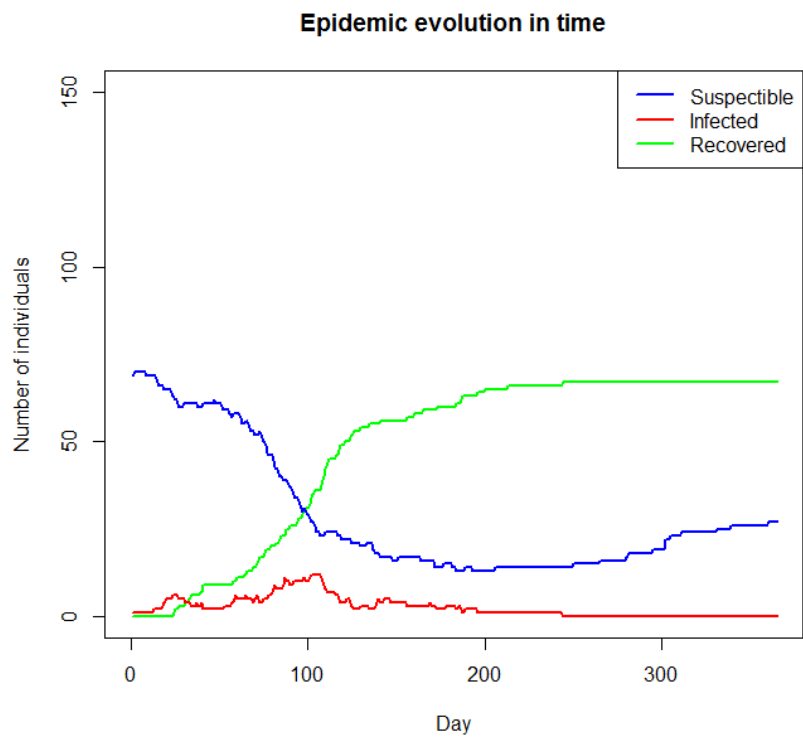


Rysunek 5.3: Początkowa dystrybucja populacji. Chora jednostka (czerwona) odseparowana od osobników podatnych (białych)

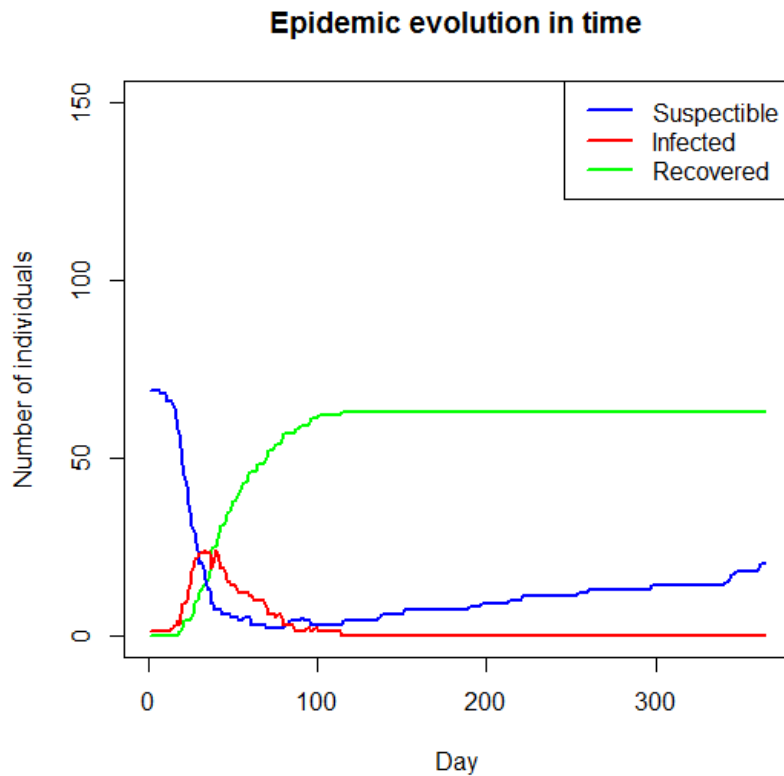
przypadku istnienia pojedynczej, prostej ścieżki transportowej rozwój epidemii, tak jak w poprzednim przypadku, bardzo przypominał ten bez preferowanych ścieżek. Szanse na rozwiniecie się epidemii były niewiele wyższe, ponieważ występowała ona średnio w 2 na 10 przypadków. Najczęściej rozwój epidemii obserwowano w przypadku zastosowania trzeciego wzorca ścieżek, zakładającego przecinanie się 2 często uczęszczanych dróg. W tym scenariuszu epidemia pojawiała się w 6 na 10 prób. Ponownie zostały potwierdzone wyniki z poprzedniego etapu, gdzie gęstsza sieć komunikacyjna spowodowała szybszą transmisję choroby, większą liczbę zainfekowanych osobników i wcześniejsze wygaśnięcie epidemii. Rezultaty tych symulacji przedstawiają wykresy na rys. 5.4.



(a)



(b)



(c)

Rysunek 5.4: Rozwój epidemii w czasie: a) dla wzorca 1 b) dla wzorca 2 c) dla wzorca 3

5.3. Testy statystyczne

W celu weryfikacji założeń na temat wpływu sieci komunikacyjnych na charakter rozwoju epidemii, wykonano testowanie statystyczne w oparciu o dane uzyskane z dwudziestu symulacji przeprowadzonych dla każdego z zaproponowanych wzorców ścieżek. Otrzymane dane składały się z 365 wpisów zawierających informacje o liczbie poszczególnych stanów komórek w kolejnych jednostkach czasu symulacji. Dane pochodziły z rozkładu normalnego, ale nie wykazywały równości wariancji, dlatego posłużono się testem Welcha-Aspina przyjmując za poziom ufności $\alpha = 0,05$. W pierwszej kolejności sprawdzono hipotezę zakładającą równość średniego czasu trwania epidemii (tj. od pierwszego zachorowania do całkowitego wyeliminowania patogenu z populacji) przy alternatywie, że jest mniejszy w próbie o bardziej gęstej infrastrukturze drogowej:

H_0 : Średni czas trwania epidemii jest taki sam dla obu prób;

H_1 : Średni czas trwania epidemii jest mniejszy w przypadku większej liczby dróg.

Test Welcha-Aspina nie wykazał podstaw do odrzucenia hipotezy zerowej dla wzorców 1 i 2, natomiast pomiędzy średnimi w próbach dla wzorców 1 i 3 oraz 2 i 3 wystąpiły wysoce istotne różnice statystyczne, co potwierdza założenie, że występowanie węzłów komunikacyjnych przyspiesza rozwój epidemii. Sprawdzono również, czy występowanie preferowanych dróg transportu zwiększa liczbę zachorowań. W tym celu porównano ze sobą średnie sumarycznej liczby osobników, które przeszły chorobę, jak również najwyższej odnotowanej liczby zainfekowanych jednostek w poszczególnych symulacjach. Testowano hipotezę:

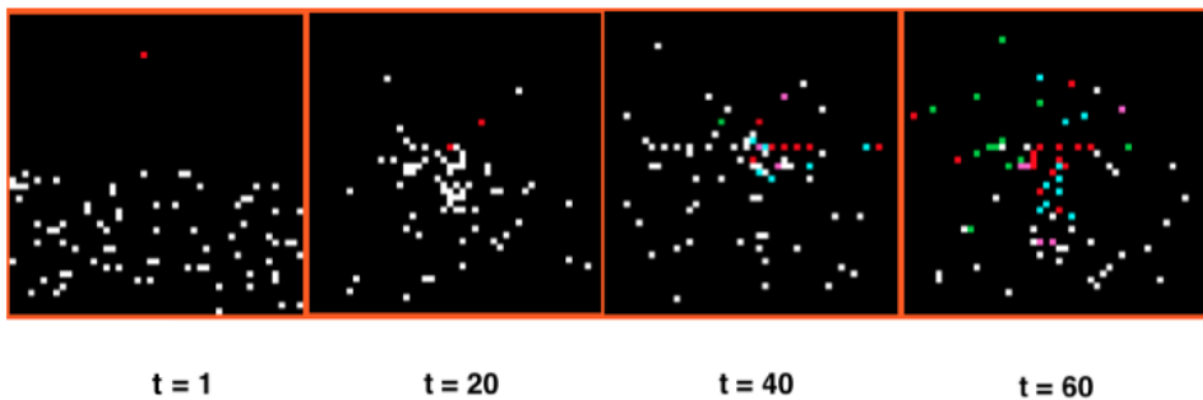
H_0 : Średnia liczba zachorowań jest taka sama dla obu prób;

H_1 : Średnia liczba zachorowań jest mniejsza w przypadku większej liczby dróg.

Podobnie jak w poprzednim przypadku test Welcha-Aspina nie wykazał różnic pomiędzy średnimi w próbach dla wzorców 1 i 2, ale pomiędzy wzorcami 1 i 3 oraz 2 i 3 wystąpiły różnice wysoce istotne, zatem również i w tym wypadku potwierdzono wpływ występowania węzłów komunikacyjnych na transmisję choroby.

6. Wnioski

Modele epidemiologiczne oparte na automatach komórkowych, które są w stanie przewidzieć stan każdej jednostki z osobna w zależności od działających na nią czynników środowiskowych, mogą okazać się niezwykle użytecznym narzędziem do symulacji ze względu na ich zdolność do uwzględniania bardzo szczegółowych danych i istotnych czynników mających wpływ na jednostkę. W niniejszej pracy zaproponowano model SEIAR uwzględniający osobników w okresie utajenia patogenu, a także tych z czynnikiem chorobotwórczym w trakcie inkubacji, którzy mimo nie wykazywania objawów odgrywają ważną rolę w zarażaniu podatnych jednostek. Należy jednak zauważyć, że użytkownikowi dana jest możliwość samodzielnego ustawienia wszystkich parametrów z osobna, co w rezultacie pozwala mu na wykluczenie z symulacji poszczególnych stanów, tak że z powodzeniem możliwe jest przeprowadzanie symulacji w oparciu o inne modele, takie jak SEIR. Implementacja algorytmu do wyszukiwania najkrótszych ścieżek umożliwia dodatkowo uwzględnienie tak istotnych czynników dla transmisji choroby jak istnienie głównych sieci tranzytowych czy barier geograficznych, co może wpływać na dokładniejsze i bardziej realistyczne analizy uwzględniające zachowanie ludzi w ruchu. Przedstawione wyniki dotyczą konkretnego zestawu parametrów przedstawionych w tabeli 5.1 i potwierdzają, że obecność preferowanych tras ruchu ma wpływ na przebieg epidemii. Wyższy odsetek zarażeń, którego wzrost widoczny jest wraz ze wzrostem gęstości połączeń jest wynikiem gromadzenia się większej liczby osobników w miejscach głównych węzłów komunikacyjnych. W tych obszarach, gdzie jednostki podróżujące w różnych kierunkach przecinają swoje drogi następuje znaczne lokalne zagęszczenie (patrz rys. 6.1), co wpływa na większe prawdopodobieństwo zarażeń między osobnikami. Z tego względu epidemia miała dużo szybszy i gwałtowniejszy przebieg, kiedy dwie trasy krzyżowały się ze sobą, aniżeli w przypadku istnienia jednej prostej drogi.



Rysunek 6.1: Koncentracja jednostek w miejscu węzła komunikacyjnego

Oznaczenie kolorów:

białe — *susceptible*, czerwone — *infected*, niebieskie — *exposed*,

fioletowe — *asymptomatic*, zielone — *recovered*

Bibliografia

- [1] Ahmed E., Agiza H. N. [1998]: *On modeling epidemics. Including latency, incubation and variable susceptibility*, Physica A 253, 347–352.
- [2] Bernoulli D. [1766]: *Essai d'une nouvelle analyse de la mortalite causee par la petite verole*, Mem Math Phy Acad Roy Sci Paris.
- [3] Chen T., Huang Y., Liu R., Xie Z., Chen S., Hu G. [2017]: *Evaluating the effects of common control measures for influenza A (H1N1) outbreak at school in China: A modeling study*, PLoS One, 12(5)
- [4] Praca zbiorowa: *Encyklopedia*, Wydawnictwo Naukowe PWN, <http://encyklopedia.pwn.pl/haslo/epidemia;3898232.html> (25.04.2017).
- [5] Kermack O., McKendrick A. G. [1927]: *Contributions to the mathematical theory of epidemics, part I*, Proc. R. Soc. Edin. A, 115, 700–721.
- [6] Lorenz E. [2000]: *The butterfly effect*, WORLD SCIENTIFIC SERIES ON NONLINEAR SCIENCE SERIES A, 39, 91–94.
- [7] López L., Burguener G., Giovanini L. [2014]: *Addressing population heterogeneity and distribution in epidemics models using a cellular automata approach*, BMC Research Notes, 7:234.
- [8] Malarz K.: *Automaty komórkowe, notatki do wykładu*, <http://www.zis.agh.edu.pl/ak/ca.pdf> (25.04.2017).
- [9] Papadimitriou C. H.: *Złożoność obliczeniowa*, Wydawnictwa Naukowo-Techniczne, 2002, 182–192.
- [10] Scott S., Duncan C. J. [2001]: *Biology of Plagues: Evidence from Historical Populations*, Cambridge University Press.
- [11] Shapiro M., Delgado-Eckert E. [2012]: *Finding the probability of infection in an SIR network is NP-Hard*, Math Biosci., 240(2): 77–84.
- [12] Sirakoulis G. Ch., Karafyllidis I., Thanailakis A. [2000]: *A cellular automaton model for the effects of population movement and vaccination on epidemic propagation*, Ecol. Model. 133, 209–223.

- [13] Stefano B. D., Fuks H., Lawniczak A. T. [2000]: *Object-oriented implementation of CA/LGCA modelling applied to the spread of epidemics*, In 2000 Canadian Conference on Electrical and Computer Engineering, vol. 1, IEEE, 26–31.
- [14] The Ecology Global Network, <http://www.ecology.com/birth-death-rates/> (25.04.2017).
- [15] Von Neumann J., Burks A. W. [1966]: *Theory of self-reproducing automata*. *IEEE Transactions on Neural Networks*, 5(1), 3–14.
- [16] White S. H., del Rey A. M., Sánchez G. R. [2007] *Modeling epidemics using cellular automata*, *Appl Math Comput* 186: 193–202.
- [17] White S. H., del Rey A. M., Sánchez G. R. [2009]: *Using cellular automata to simulate epidemic diseases*, *Applied Mathematical Sciences*, 3(20), 959–968.
- [18] Wolfram S. [1984]: *Universality and complexity in cellular automata*, *Physica D: Nonlinear Phenomena*, 10(1–2), 1–35.
- [19] Wikipedia, the free encyclopedia, https://en.wikipedia.org/wiki/Rule_110 (08.06.2017).
- [20] Wróblewski P. [2010]: *Algorytmy, struktury danych i techniki programowania*, Helion, wyd. IV: 224–225.

Spis rysunków

1.1	Schemat funkcji stanu (F)	12
1.2	Przykładowy wynik reguły 110 (źródło: Wikipedia [19])	14
1.3	Przykładowe rodzaje siatek	15
1.4	Typy sąsiedztwa	15
1.5	Etapy infekcji wirusowej	17
2.1	Schemat modelu SIR	19
2.2	Schemat modelu SEIAR	20
3.1	Model SEIAR z prawdopodobieństwami przejść między stanami	25
3.2	Algorytm infekcji	29
3.3	Algorytm ekspozycji	29
3.4	Algorytm spontanicznych zachorowań	30
3.5	Algorytm powrotu do zdrowia	31
3.6	Algorytm śmierci wywołanej chorobą	32
3.7	Algorytm naturalnych narodzin i śmierci	33
3.8	Algorytm ruchu	34
4.1	Wzorce transportu	40
5.1	Liczba zarażonych jednostek	43
5.2	Rozwój epidemii w czasie	45
5.3	Początkowa dystrybucja populacji	46
5.4	Rozwój epidemii w czasie	48
6.1	Koncentracja jednostek w miejscu węzła komunikacyjnego	51

Spis tabel

1.1	Reguła 110	13
2.1	Podsumowanie modeli	22
3.1	Prawdopodobieństwa przejść pomiędzy stanami	27
5.1	Parametry symulacji	41