

# Modelowanie epidemii chorób zakaźnych za pomocą automatów komórkowych

Dominika Bocheńczyk, Antonina Kuś, Piotr Magiera

15 czerwca 2022

## 1 Model McKendricka-Kermacka

### 1.1 Interpretacja modelu

Wybrany przez nas model reprezentuje sieć powiązań w lokalnej społeczności (np. firmie, osiedlu). Każda komórka reprezentuje pojedynczą osobę, a jej stan oznacza odpowiednio:

- S - podatna na infekcję,
- I - zainfekowana,
- R - odporna na infekcję,
- V - zaszczepiona.

Sąsiedzi danej komórki to osoby, które mogą zainfekować bezpośrednio daną jednostkę (np. najbliżsi współpracownicy, sąsiedzi).

### 1.2 Parametry

Powyższy model opiera się na automacie komórkowym wykorzystującym sąsiedztwo Moore'a o promieniu równym jeden oraz dwóch parametrach -  $p$  oraz  $q$  - determinujących odpowiednio prawdopodobieństwa zakażenia i wyzdrowienia:

$$P_{\text{infect}} = 1 - (1 - p)^n,$$

$$P_{\text{recover}} = q.$$

Liczba sąsiadów symbolizowana przez  $n$  wynosi w naszym przypadku 8. Poniższe pseudokody prezentują pojedynczą iterację w danym modelu.

### 1.3 SIR asynchroniczny

```
for cell in cells:
    if not cell.used and cell.site == I:
        if rand(0, 1) < p_recover:
            this.site = R
        else:
            pick random neighbor N
            if not N.used and N.site == S and rand(0, 1) < p_infect:
                N.site = I
                N.used = True
    cell.used = True
```

## 1.4 SIS asynchroniczny

```
for cell in cells:
    if not cell.used and cell.site == I:
        if rand(0, 1) < p_recover:
            this.site = S
        else:
            pick random neighbor N
            if not N.used and N.site == S and rand(0, 1) < p_infect:
                N.site = I
                N.used = True
            cell.used = True
```

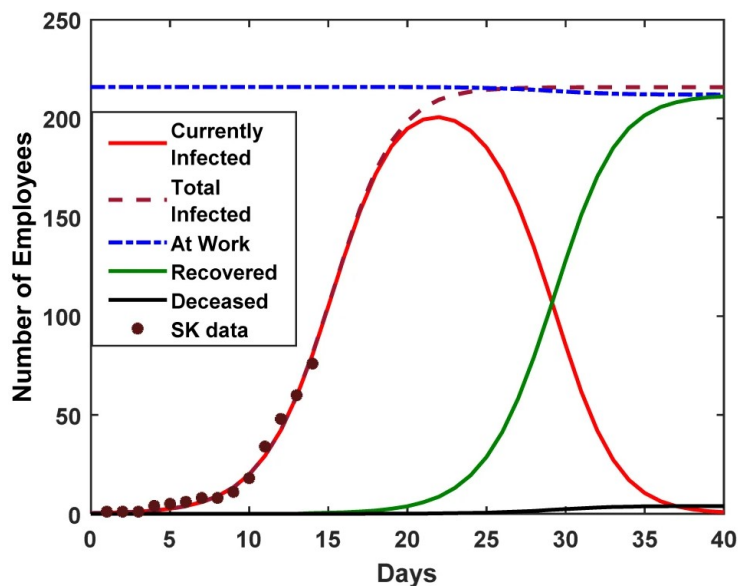
## 1.5 SIRV asynchroniczny

```
for cell in cells:
    if not cell.used and cell.site == I:
        if rand(0, 1) < p_recover:
            this.site = R
        else:
            pick random neighbor N
            if not N.used and N.site == S and rand(0, 1) < p_infect:
                N.site = I
                N.used = True
            cell.used = True
    if cell.site == V:
        if vaccineDay - currDay < vaccineDuration:
            cell.site = S
    if cell.site == S:
        if rand(0, 1) < p_vaccine:
            this.site = V
            this.vaccineDay = currDay
```

## 2 Przykłady działania symulacji

Z uwagi na problemy ze znalezieniem danych dotyczących rozprzestrzeniania się chorób dla mniejszych społeczności (statystyki są prowadzone głównie dla np. województw, stanów, państw) zvalidowaliśmy nasz model tylko dla dwóch przypadków:

- COVID-19 - na podstawie danych z pewnym telefonicznym centrum obsługi klienta (nazwa nie została podana) w Korei Południowej. Oryginalny wykres prezentuje się następująco:

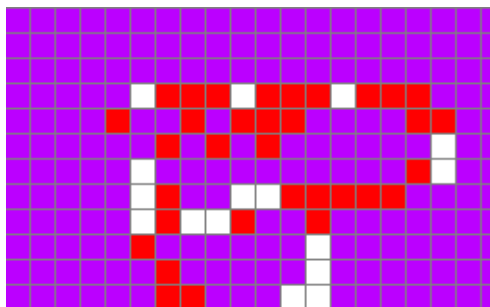


Rysunek 1: COVID-19 - oryginalny wykres

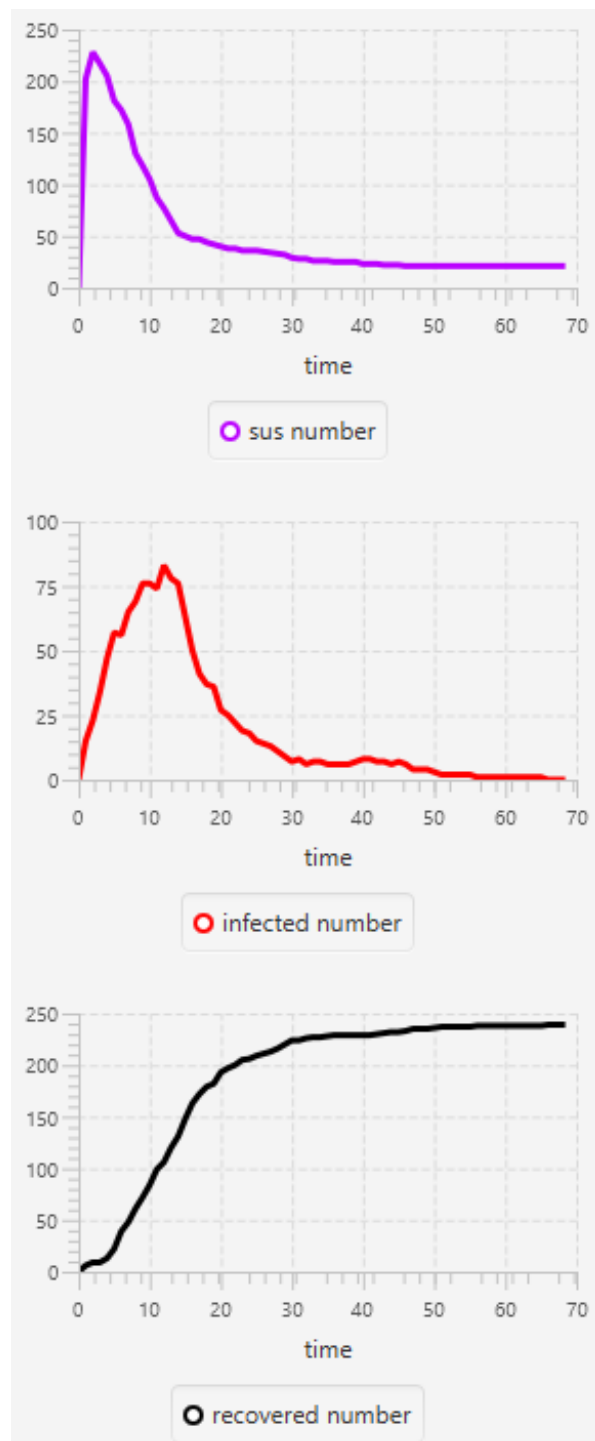
Po kilku próbach symulacji udało uzyskać się nam podobny wykres dla parametrów o wartościach

$$p = 0.5,$$

$$q = 0.2.$$

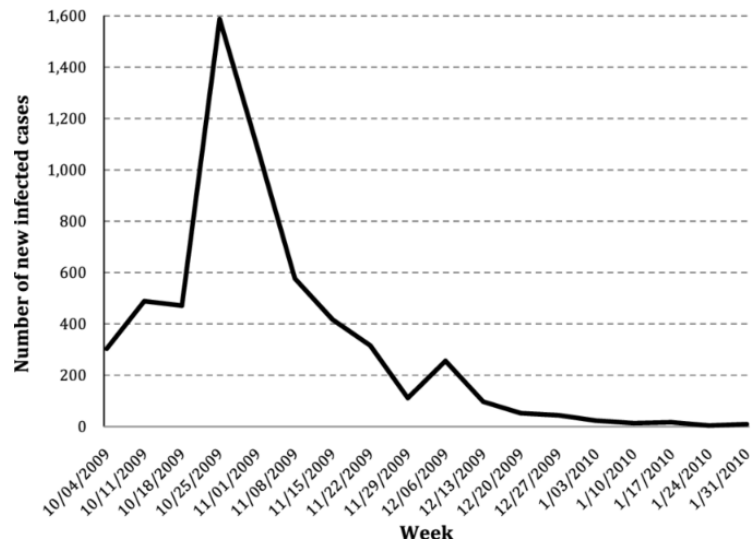


Rysunek 2: COVID-19 - warunki początkowe naszej symulacji



Rysunek 3: COVID-19 - wykres uzyskany na podstawie naszej symulacji

- H1N1 - na podstawie danych z badania nt. wybuchu epidemii H1N1 w stanie Arizona z 2009 roku.

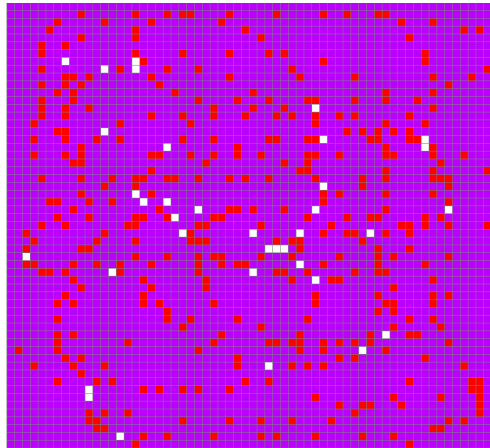


Rysunek 4: H1N1 - oryginalny wykres

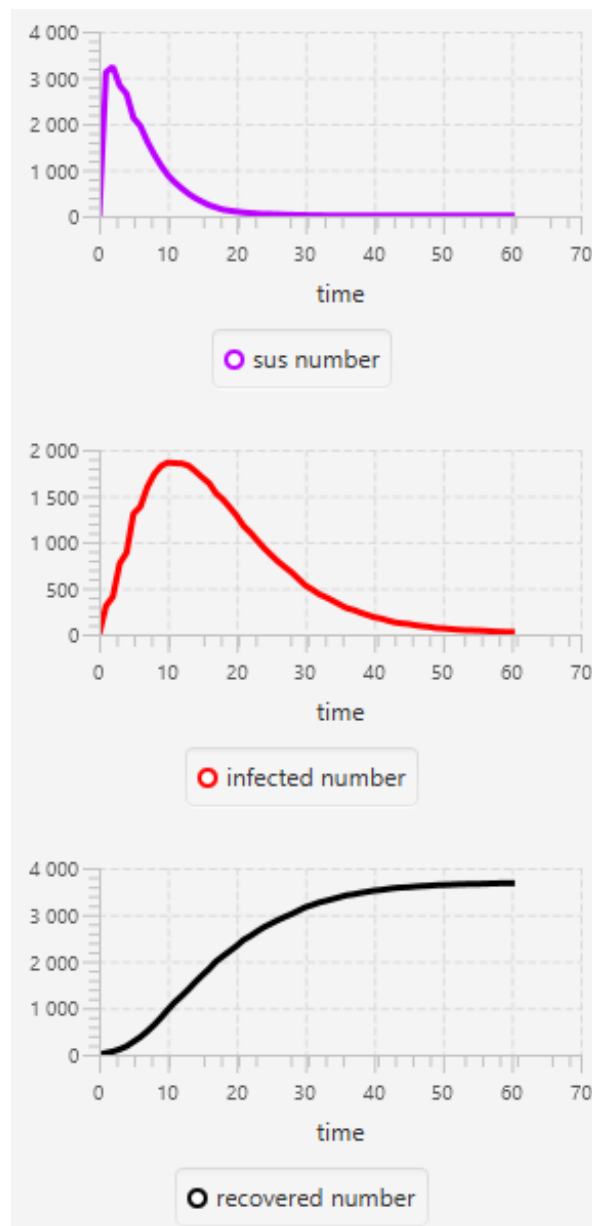
Po kilku próbach symulacji udało uzyskać się nam podobny wykres dla parametrów o wartościach

$$p = 0.9,$$

$$q = 0.1.$$



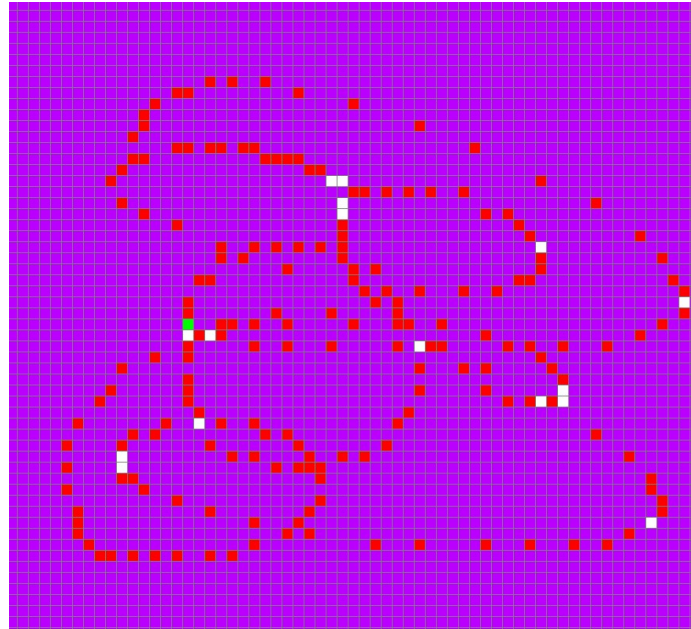
Rysunek 5: H1N1 - warunki początkowe naszej symulacji



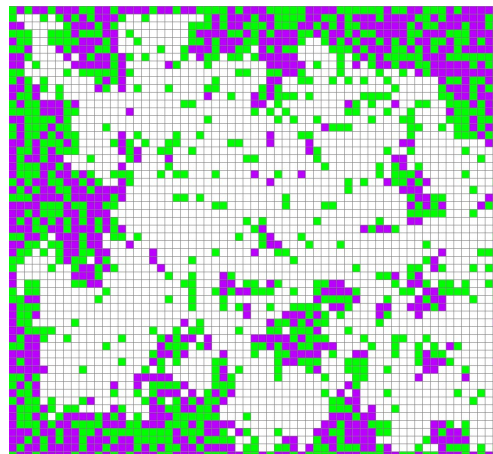
Rysunek 6: H1N1 - wykres uzyskany na podstawie naszej symulacji

### 3 Wnioski

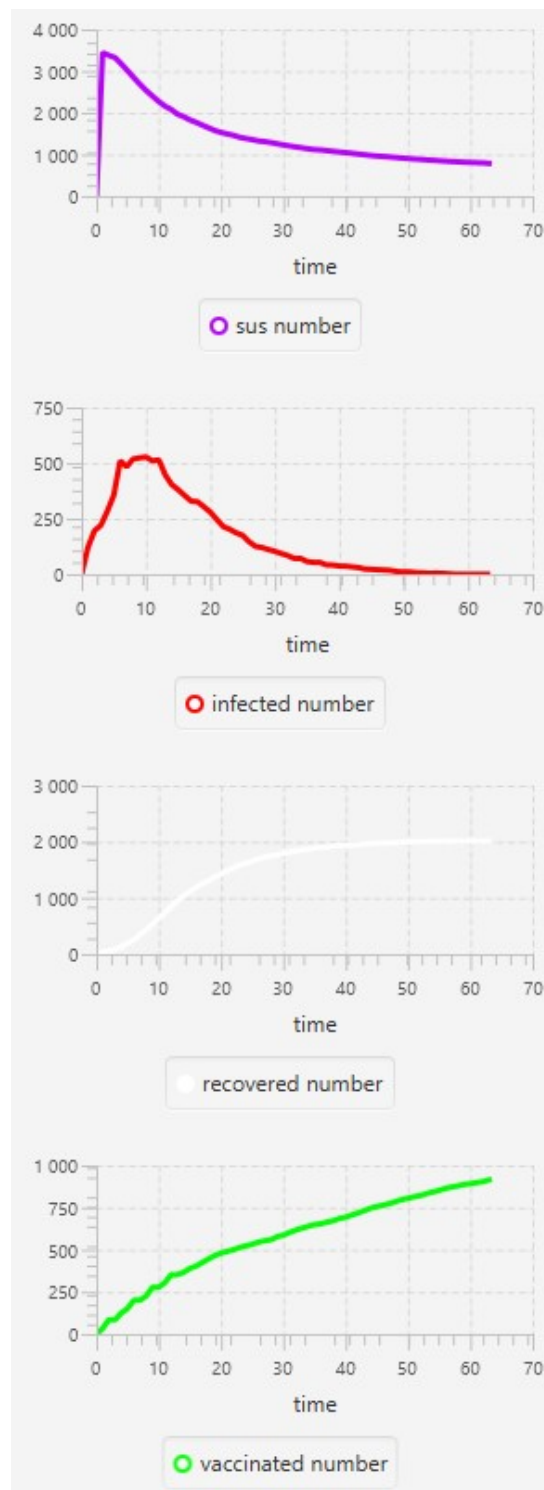
- Obserwując kolejne symulacje zauważyć można, że szczepionki w modelu SIRV chronią nie tylko osoby zaszczepione (V), ale także niektóre osoby podatne na infekcję (S) co widać na poniższych grafikach.



Rysunek 7: Szczepionki - warunki początkowe symulacji



Rysunek 8: Szczepionki - stan końcowy symulacji



Rysunek 9: Szczepionki - wykres uzyskany na podstawie naszej symulacji



- Wynik symulacji w dużym stopniu zależy od stanu początkowego komórek.
- Zwiększenie parametru  $q$  wydłuża czas trwania epidemii, ale przebiega ona łagodniej.
- Zwiększenie parametru  $p$  skraca czas trwania epidemii w modelu SIR, ale jest ona dużo bardziej gwałtowna.

## 4 Implementacja

Do zaimplementowania modelu został użyty język Java.

Repozytorium z kodem źródłowym: <https://github.com/piotmag769/PandemicSimulation>

## 5 Bibliografia

[https://en.wikipedia.org/wiki/Compartmental\\_models\\_in\\_epidemiology](https://en.wikipedia.org/wiki/Compartmental_models_in_epidemiology)  
<http://csc.ucdavis.edu/~chaos/courses/nlp/Projects2008/SharonChang/Report.pdf>  
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7127728/>  
[https://en.wikipedia.org/wiki/Epidemic\\_models\\_on\\_lattices](https://en.wikipedia.org/wiki/Epidemic_models_on_lattices)  
<https://www.nature.com/articles/s41370-022-00411-2#ethics>  
[https://www.researchgate.net/publication/257548978\\_Modeling\\_dynamics\\_of\\_an\\_influenza\\_pandemic\\_with\\_heterogeneous\\_coping\\_behaviors\\_Case\\_study\\_of\\_a\\_2009\\_H1N1\\_outbreak\\_in\\_Arizona](https://www.researchgate.net/publication/257548978_Modeling_dynamics_of_an_influenza_pandemic_with_heterogeneous_coping_behaviors_Case_study_of_a_2009_H1N1_outbreak_in_Arizona)