

Processing を用いた感染シミュレーターの作成

5年C組 笠谷 悠豪

5年C組 坂本 悠

指導教員 高森 智子

1. 概要

Processing を用いてライフゲームの仕組みを利用した感染シミュレーターを作成した。その後、新型コロナ感染症と条件を等しくしてシミュレーションを行い、その結果と実際の感染状況を比較してシミュレーターの精度の評価を行った。

2. 研究理由

新型コロナウイルスの感染拡大の際、身近に使える感染シミュレーターがあれば、どのような感染が予想されるかを個人で知ることができ、感染症に対する評価をできると考えた。そこで、動作が軽く、誰にでも使用しやすいシミュレーターの開発を行うことにした。

3. 研究内容

3-1 Processing について

Processing とはプログラミング言語の一種である。Java を単純化しておりグラフィックに特化している。

Processing を用いた理由としては、視覚的な表現が行いやすい点と様々な OS

(Windows・Mac・Android 等) に対応している点などが挙げられる。

3-2 シミュレーションについて

今回のシミュレーターは、人をマスで表しており、そのマス目を正方形に生成している。人は以下の3つの中のいずれかの状態を取る。

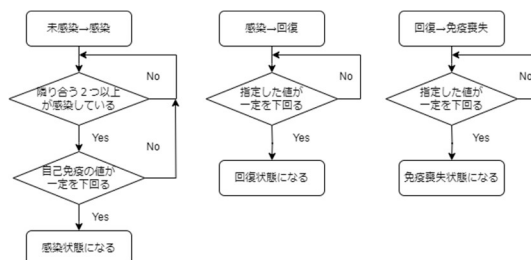
未感染者	感染しておらず免疫がない
感染者	感染している
回復者	回復し免疫を持っている

この3つの状態はそれぞれ対応した色に変化するようになって入る。

また、作成するパラメータとしては次の6つを設定している。

- ・マスクの大きさ
- ・初期の感染者の割合
- ・回復速度
- ・免疫喪失の速度
- ・世代変更の間隔
- ・自己免疫

またプログラムは以下のフローチャートに従って作成した。

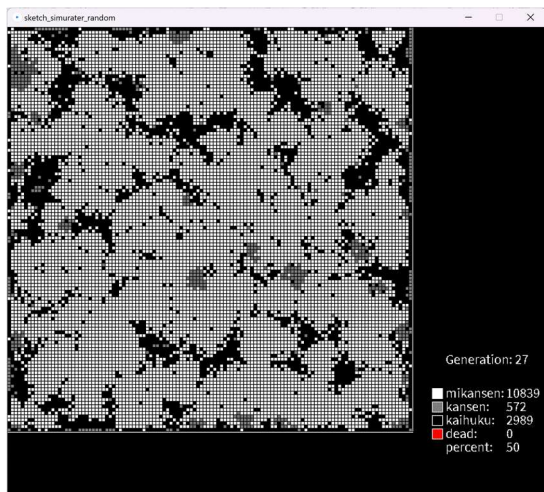


回復速度、免疫喪失の速度については指定した範囲で乱数を取り、指定した値を超えたら状態が変更されるという仕様になっ

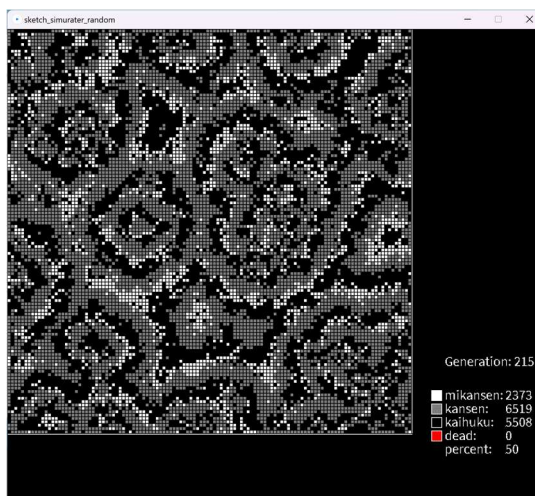
ている。また、自己免疫とは感染する確率を表したものであり、この値も同様に指定した値の中で乱数を取り、その乱数が指定した値を下回っていた際に感染するという仕様になっている。

3-3 シミュレーションの様子

先行研究のシミュレーターは各パラメーターが微分方程式で表されており、その確率に応じて他に感染していくという仕様だったのに対して、作成したシミュレーションには、隣り合う感染者の数に応じて感染するというようなわかりやすいアルゴリズムとなっている。また、それぞれの人の状況が色で区別されているため全体の変化が視覚的にわかりやすいという利点を得ることができた。(図1)しかし、第0世代においての感染者がランダムに生成されることによりシミュレーション結果に大きな差が出ることもあった。それに加え、ライフゲームの原理を利用しているため規則的な変化が生まれる場合もあった(図2)。



(図1) シミュレーションの様子



(図2) 円状に広がる感染者

4. シミュレーション結果と実際の感染状況との比較

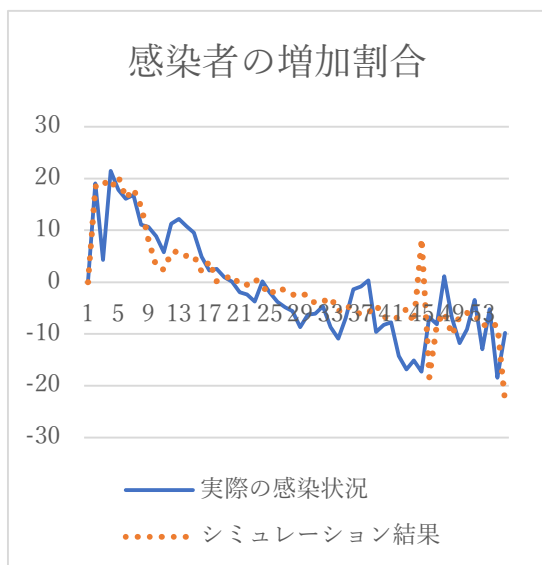
今回は日本での第一波(2020年第13週~2020年第20週)での新型コロナ感染者の数値と感染シミュレーターの第1世代~第55世代の比較を行った。下の表は7日間の移動平均の値である。

2020/3/22	93.14286	2020/4/10	533.7143	2020/4/29	213.4286
2020/3/23	110.8571	2020/4/11	523.2857	2020/4/30	195.7143
2020/3/24	115.5714	2020/4/12	510.8571	2020/5/1	180.4286
2020/3/25	140.4286	2020/4/13	491.7143	2020/5/2	154.7143
2020/3/26	165.4286	2020/4/14	492.2857	2020/5/3	128.5714
2020/3/27	192	2020/4/15	482.1429	2020/5/4	109.1429
2020/3/28	224.2857	2020/4/16	463.5714	2020/5/5	90.28571
2020/3/29	249.1429	2020/4/17	440.8571	2020/5/6	84.14286
2020/3/30	275.7143	2020/4/18	416.1429	2020/5/7	77.28571
2020/3/31	300.1429	2020/4/19	379.8571	2020/5/8	78.14286
2020/4/1	317.4286	2020/4/20	355.8571	2020/5/9	72.71429
2020/4/2	353.1429	2020/4/21	334	2020/5/10	64.14286
2020/4/3	396.1429	2020/4/22	318.4286	2020/5/11	58.28571
2020/4/4	438.7143	2020/4/23	290.7143	2020/5/12	56.28571
2020/4/5	480.5714	2020/4/24	258.8571	2020/5/13	49
2020/4/6	504	2020/4/25	240.8571	2020/5/14	46.42857
2020/4/7	515.1429	2020/4/26	237.4286	2020/5/15	37.85714
2020/4/8	528.2857	2020/4/27	235.4286	2020/5/16	34.14286
2020/4/9	533.2857	2020/4/28	236.1429		

(表1) 日本での第一波における感染者数の推移の移動平均

新型コロナ感染症において、発症2日前~発症10日後まで感染させることが可能であり、免疫は4か月から8か月持つと仮定しこの条件でシミュレーションを行った。なお、感染による死亡については、考えないものとしてシミュレーションを行っ

ている。結果は下のグラフのようになった。



(グラフ 1) 感染者の比較結果

グラフ 1 より実際の感染状況とシミュレーション結果はかなり近い値を取った。

5. 考察

シミュレーション結果と実際の感染状況がかなり近くなったため、作成したシミュレーションは十分な精度であると考えられる。この結果は、回復速度、免疫喪失のランダム化を実装する前のシミュレーターの誤差がかなり大きかったため、これらの仕様によって得られたものだと考えられる。

一方で、第 0 世代における結果の差異については今後改善方法を考えていく必要がある。

6. 今後の展望

シミュレーターとしての精度をさらに上げるために、感染におけるワクチンやクラスターなどの外部的要因の導入や感染者との距離に応じた感染率を変化させるなどの仕様も作成していきたい。

7. 参考文献

西浦 博・稲葉 寿(2006). 「感染症流行の予測：感染症数理モデルにおける定量的課題」. 『統計数理』. 54(2). 461-480

坂本 啓法(2017) 「SIR 感染症モデルのシミュレーター」

<https://sakapon.wordpress.com/2017/12/19/epidemic-simulator/>
(2023-04-26)

厚生労働省, 新規陽性者数の推移 (日別), 2023-05-08

<https://www.mhlw.go.jp/stf/covid-19/open-data.html>
(2024-1-24)

国立感染症研究所感染症疫学センター, 日本における新型コロナウイルス感染症の流行波ごとの性別・年齢的特徴の疫学的検討, 2022-12-23

<https://www.niid.go.jp/niid/ja/typhim/iasr-reference/2605-related-articles/related-articles-514/11696-514r01.html>
(2024-1-31)

土谷隆. “新型コロナウイルス感染症の広がりに関する一考察.” 2020-05-30 2020. 2020-05-30 (2020).