

武汉肺炎传播的多种数学模型

大模头，微信公众号“数学模型”

2020 年 01 月 29 日



- 本文档内可能含有动画，使用 Adobe Reader 打开才能播放。
- 由于微信公众号限制评论，欢迎在知乎专栏中评论本文。

1 引言

2019 年 12 月在武汉市发现有人感染了一种新型冠状病毒肺炎，这种病毒被命名为 2019-nCoV，这种新型冠状肺炎也被称为“武汉肺炎”。随后的几十天内，该肺炎迅速蔓延至全国其它省份及全球其它国家。截至 2020 年 1 月 29 日，全球已经确诊近 6,140 例，其中中国大陆占 6,051 例，死亡 132 例 [1]。

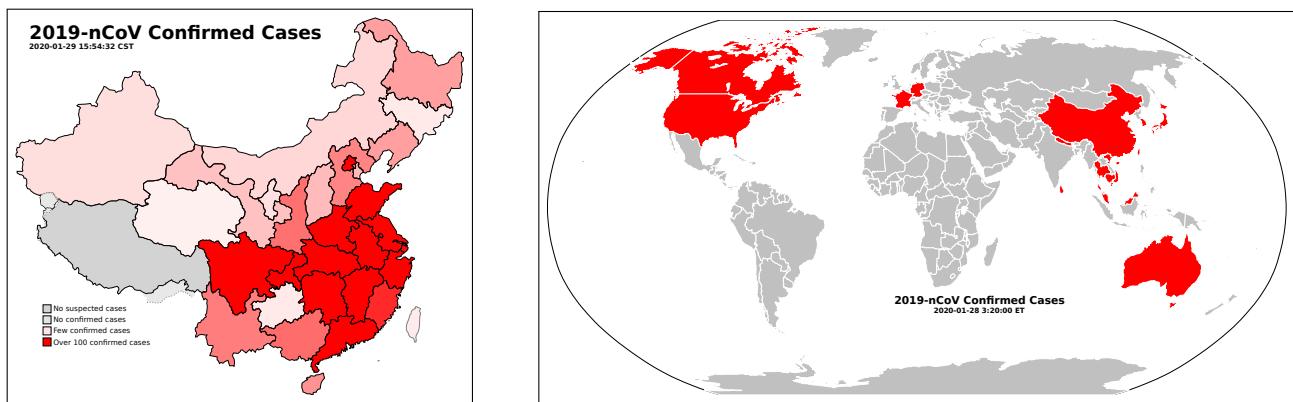


图 1：新型冠状病毒肺炎全国分布图（1 月 29 日）和世界分布图（1 月 28 日）[2]

一周内网络出现了大量研究新型冠状病毒传播的数学模型 [3–6]，大模头参考了这些模型，针对武汉新型冠状病毒，分别构建了回溯传播模型、SIR 模型、SEIR 模型和元胞自动机模型。模型和模拟结果表明：

- 武汉在 2020 年 1 月 18 日感染人数可能就已经超过 2700 人。
- 如果不加控制，武汉市疫情在 3 月上旬左右达到高峰，4 月中旬接近尾声，最终死亡人数将超过 32 万人。
- 以当前政府的管控力度，预计武汉累计感染人数将控制在 25000 人以内，累计死亡人数将控制在 720 人以内。
- 新型冠状病毒基本传染数约为 3.6，大于非典和埃博拉。

以上结论是大模头基于有限的数据和理想的模型得到的，不代表真实情况，请各位模友自行判断可信性。

2 回溯传播模型

几乎所有新病毒造成的传染病都会存在疫源地遗漏病例的情况，随着诊断方法的发展和分发，病例的确诊会得到优化，而且监管也会得到加强。而那些尚未受到影响的国家已经处于警戒状态，且更有可能发现偶发输入的病例。因为我们知道每天从武汉飞往中国大陆以外目的地的人数，我们便能计算任何 2019-nCov 感染病例无意间飞往海外的可能性。通过回溯海外检出病例可以评估武汉疫情的可能真实规模。我们做出如下假设：

- 多国对来自武汉旅客增设入境筛查主要开始于 1 月 15 日 [7]，假设截至 1 月 16 日前未病发旅客能够自由离境、且随机分布在离境与未离境人口中。
- 假设从武汉飞往中国大陆以外目的地的病例都在其目的地被确诊。很可能不是这样。如果某些国家漏掉了少量病例，将导致我们低估武汉肺炎病例的数量。

假设感染总人数为 M ，其中境外确诊病例为 m ，因此任意出境者在目的被诊断为病例的概率为

$$p = \frac{m}{M} \quad (2.1)$$

其中 p 由平均潜伏期内武汉市人口出境比例给出

$$p = \frac{n}{N} \cdot T \quad (2.2)$$

其中 n 为每天从武汉出境的人数， N 为武汉总人数， T 为从感染到确诊的平均时间，可由以下公式估计：

$$T = \text{平均潜伏期} + \text{发病到确诊的平均时间} \quad (2.3)$$

根据以上三式可以反推出感染总人数 M 。

2.1 参数

为了应用上述模型反推出感染总人数，我们需要估计出以下参数：

- 武汉机场覆盖人口数（以武汉市人口计算） $N = 1300$ 万 [8]。
- 新型肺炎病人从感染到发现的平均时间为 $T = 10$ 天，包括 5-6 天潜伏期 [9] 和发病到确诊 4-5 天（泰国和日本发现的病例分别在发病后 3 天和 7 天住院 [10, 11]）。
- 过去两个月，武汉的国际旅行总量为每天 $n = 3300$ 人次。这是根据 2018 年国际航空运输协会（IATA）的数据 [12] 估算的。
- 截止 2020 年 1 月 12 日为止，境外报告了 $m = 3$ 例感染者。截止 2020 年 1 月 18 日为止，境外报告了 $m = 7$ 例感染者 [13]。

2.2 结果

根据以上模型和参数，我们可以估算截止 2020 年 1 月 18 日武汉因新病毒引起症状的人数应为

$$M = \frac{m \cdot N}{n \cdot T} = \frac{7 \times 1.3 \times 10^7}{3300 \times 10} = 2758 \quad (2.4)$$

而 2020 年 1 月 18 日国内报道的确诊病例只有 136 人，如果模型计算是可信的，那么报道出来的只占实际的 5%。以上估计所用的回溯传播模型主要源于英国帝国理工学院的全球传染病分析中心发布的关于武汉市内新型冠状病毒潜在病例总数的第二份报告 [3]。由于不确定因素较多，该估算存在多处局限，不确定的范围约是 1000-10000。以同样的方式可以估计出 2020 年 1 月 12 日武汉因新病毒引起症状的人数应为 1182。

3 SIR 模型

在往期的推文中，我们已经介绍过 SIR 模型，详情见《如果没有国家管控，北京鼠疫将如何漫延？》。这里我们再简单回顾一下该模型。SIR 模型中涉及三种人群：易感人群（Susceptible），感染人群（Infective）和移除人群（Removed）。易感人群指未得病者，但缺乏免疫能力，与感染者接触后容易受到感染。感染人群指染上传染病的人。移除人群是因病愈（具有免疫力）或死亡而不再参与感染和被感染过程的人。如图 2 所示， S 、 I 和 R 分别表示易感人群、感染人群和移除人群的数量。随着疫

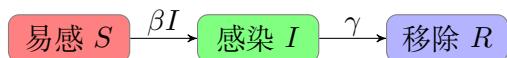


图 2: SIR 传染病模型

情的发展，易感人群逐渐被感染，其变化率为：

$$\frac{dS}{dt} = -\beta \frac{IS}{N} \quad (3.1)$$

其中 β 为感染者每天平均接触到的人数 k 及接触时传染概率 b 的乘积，即 $\beta = kb$ 。感染人群的变化率为：

$$\frac{dI}{dt} = \beta \frac{IS}{N} - \gamma I \quad (3.2)$$

其中 $\gamma = 1/C$ 为感染人群恢复或死亡的平均速率，取决于感染的平均持续时间 C 。移除人群的变化率为：

$$\frac{dR}{dt} = \gamma I \quad (3.3)$$

由于死亡滞后于感染，假设感染人群的死亡率为 d ，则 t 时刻前的累计死亡人数为

$$D(t) = d \cdot [N - S(t - C)], \quad t > C \quad (3.4)$$

3.1 参数

为了将 SIR 模型应用到武汉肺炎，我们需要对参数 β 、 γ 和 d 进行估计。因为新型冠状病毒肺炎的恢复期大约是 $C = 14$ 天，因此 $\gamma = 1/14$ 。根据报道 [14]，我们假设死亡率为 3 %。假设感染者每天平均接触人数为 $k = 5$ ，因此只需要再根据数据估计出传染概率 b 即可。考虑到武汉可能会存在漏报病例的情况，我们采用表 1 和 2 两组数据分别对 b 进行估计。以 2019 年 12 月 8 日为第 0 天，起

表 1: 政府官方公布的数据

时间	感染人数 (I)
1月19日	198
1月20日	218
1月21日	320
1月22日	478
1月23日	639

表 2: 回溯传播模型估算的数据

时间	感染人数 (I)
1月12日	1182
1月18日	2758

始感染人数为 1。我们估计参数所使用的数据是肺炎疫情早期数据，在疫情早期有 $N \approx S$ ，因此式 (3.2) 近似于

$$\frac{dI}{dt} = \beta \frac{IS}{N} - \gamma I \approx (\beta - \gamma)I \quad (3.5)$$

考虑到 $I(t=0) = 1$ 上式的解为

$$I(t) = e^{(k \cdot b - \gamma)t} \quad (3.6)$$

对表 2 和表 1 的数据分别按照式 (3.6) 拟合可得 $b = 0.04133$ 和 $b = 0.05214$ 。拟合所使用的程序见附录代码 1。

此外, 我们还可以对新型冠状病毒基本传染数 (basic reproduction number) R_0 进行估计:

$$R_0 = \frac{\beta}{\gamma} = \frac{kb}{1/D} = \frac{5 \times 0.04133 \text{ 或 } 0.05214}{1/14} = 2.9 \text{ 或 } 3.6 \quad (3.7)$$

本文估计的 R_0 与维基中给出的 2.3–5 非常吻合 [15]。基本传染数是指在没有外力介入, 同时所有人都没有免疫力的情况下, 一个感染某种传染病的人, 会把疾病传染给其他多少个人的平均数。 R_0 数字愈大, 代表流行病的控制愈难。如果 R_0 大于 1, 传染病会以指数方式散布, 成为流行病; 而 R_0 小于 1, 传染病将会逐渐消失。非典的基本传染数约为 0.85–3, 埃博拉基本传染数约 1.5–2.5。而我们估计的新型冠状病毒基本传染数 R_0 为 2.9 或 3.6, 这表明当前武汉疫情危险程度远大于非典和埃博拉。

3.2 结果

如果仅对武汉进行封城, 并不对肺炎病人的城内活动进行限制, 则模拟结果如图 3 所示, 模拟所用程序见附录代码 3。图中的第 0 天对应着发现第一个病例的时间, 即 2019 年 12 月 8 日。图 3(a) 是根据官方数据拟合的 SIR 模型结果, 从图中可以看出, 疫情 70 天左右开始集中爆发 (2 月 10 日左右), 120 天左右达到高峰 (4 月上旬), 150 天后接近尾声 (5 月上旬), 最终死亡人数将超过 30 万人。图 3(b) 是根据回溯传播模型估算数据拟合的 SIR 模型结果, 从图中可以看出, 疫情 50 天左右开始集中爆发 (1 月 20 日左右), 90 天左右达到高峰 (3 月上旬), 130 天后接近尾声 (4 月中旬), 最终死亡人数将超过 32 万人。

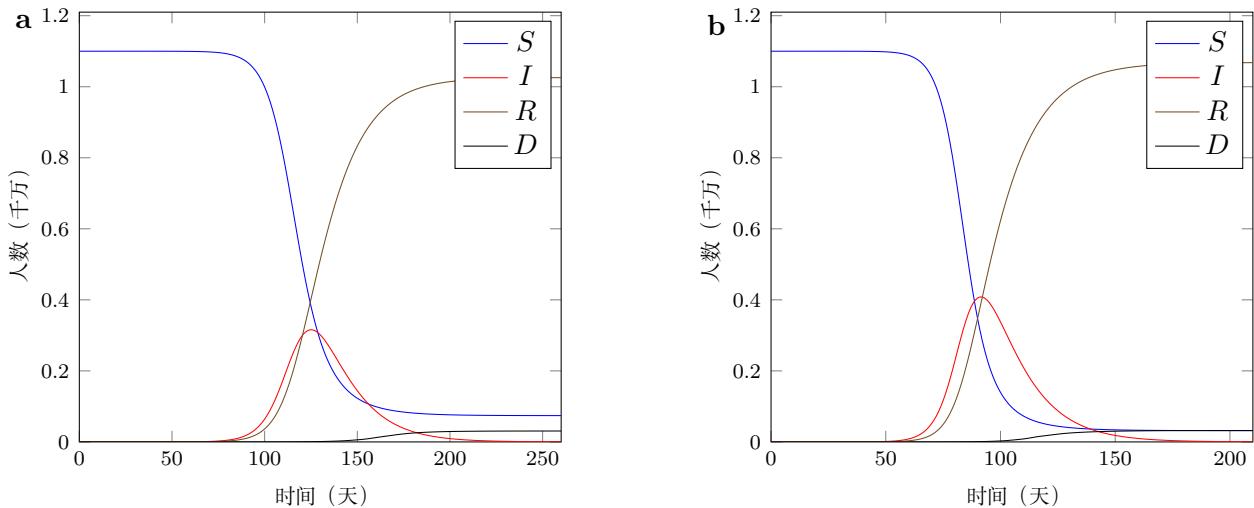


图 3: SIR 模型给出的四种人群数量随时间的变化。(a) $b = 0.04133$; (b) $b = 0.05214$ 。

实际上, 我国政府在 1 月 23 日不仅对武汉进行封城, 也对武汉内的交通进行了限制, 并关闭了各种不必要的公共场所。我们可以通过调整感染者每天平均接触到的人数 k 来模拟政府管控的力度, 图 4 显示了不同管控强度的模拟结果。模拟所用程序见附录代码 4。

采用官方数据拟合时, 如果政府管控力度非常强 (图 4(a)), 武汉累计感染人数将控制在 1500 人以内, 累计死亡人数将控制在 50 人以内; 如果政府管控力度稍弱 (图 4(c)), 武汉累计感染人数将超过 3000 人, 累计死亡人数将达到 100 人。采用回溯传播模型估算的数据拟合时, 如果政府管控力度非常强 (图 4(b)), 武汉累计感染人数将控制在 25000 人以内, 累计死亡人数将控制在 720 人以内; 如果政府管控力度稍弱 (图 4(d)), 武汉累计感染人数将超过 190 万人, 累计死亡人数将达到 5.7 万人。

通过将图 4 四种情况与当前疫情 [13] 比较, 可以发现图 4(b) 所示的结果比较接近真实情况。这说明早期武汉政府披露的感染人数可能比实际少得多, 这也在新闻报道中得到了印证 [14, 16]。同时也说明政府当前的管控力度非常强。当然, 本模型没有考虑政府进一步的措施, 比如研制新药、疫苗

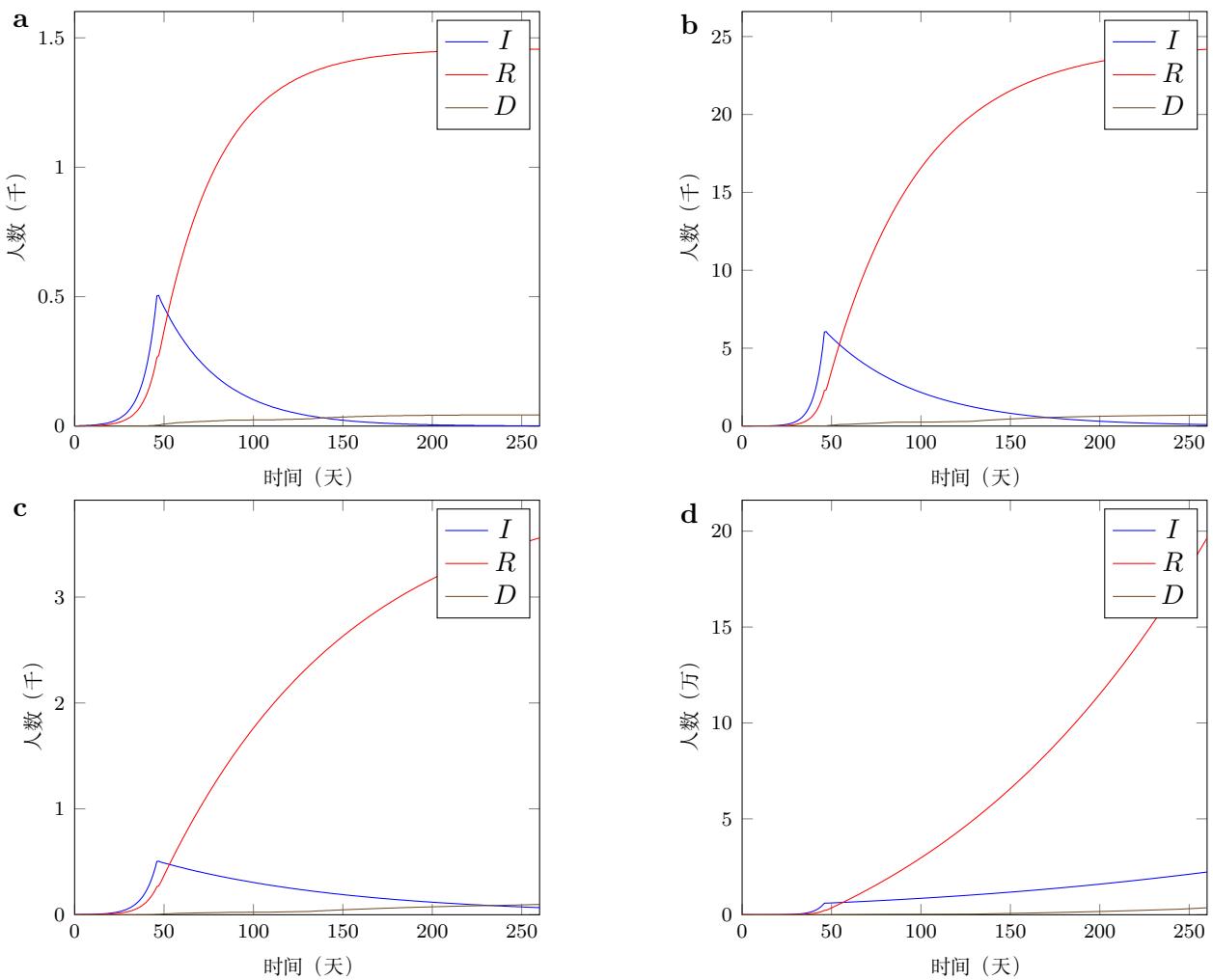


图 4: 不同管控强度的 SIR 模型给出的四种人群数量随时间的变化。(a) $b = 0.04133$, $k = 1$; (b) $b = 0.05214$, $k = 1$; (c) $b = 0.04133$, $k = 1.5$; (d) $b = 0.05214$, $k = 1.5$ 。

等。因此图 4(b) 给出的“武汉累计感染人数将控制在 25000 人以内, 累计死亡人数将控制在 720 人以内”可能会比最终的实际情况高。

4 SIER 模型

在第 3 节 SIR 模型的基础上, 本节将构建包含感染并处于潜伏期人群的 SIER 模型。SEIR 将人群分为四类: 易感人群 (Susceptible), 潜伏人群 (Enfective), 隔离 (确诊) 人群 (Infective) 和移除人群 (Removed)。

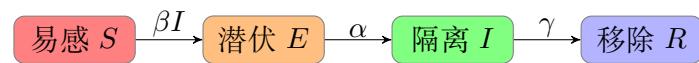


图 5: SEIR 传染病模型

随着疫情的发展, 四种人群的变化率分别为:

$$\frac{dS}{dt} = -\beta \frac{IS}{N} \quad (4.1)$$

$$\frac{dE}{dt} = \beta \frac{IS}{N} - \alpha E \quad (4.2)$$

$$\frac{dI}{dt} = \alpha E - \gamma I \quad (4.3)$$

$$\frac{dR}{dt} = \gamma I \quad (4.4)$$

其中 β, γ 的取值与上一小节的 SIR 模型相同, $\alpha = 1/T$ 为隔离速率, 由于新冠状病毒的平均潜伏期 = 7, 故 $\alpha = 1/7$ 。

4.1 结果

由于在第 3 节 SIR 模型中已经讨论过政府不同管控力度的情况, 这里不再细致讨论。图 6 是 SIER 模型的结果。SIER 已经包含了隔离的情况, 但只对发病的人进行隔离, 并没有考虑人们自发减少出行、聚会等变相的隔离。模拟所使用程序见附录中的代码 5。

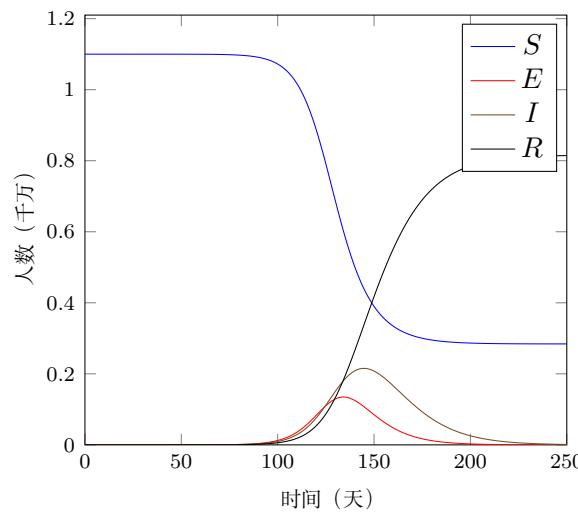


图 6: SEIR 传染病模型给出的四种人群数量随时间的变化

5 元胞自动机模型

对于像武汉肺炎这种大规模的传染病问题, 大模头并不推荐用元胞自动机模型进行模拟。元胞自动机更适合于模拟空间上的火灾问题, 详情见《澳洲变澳洲, 考拉成烤拉! 澳大利亚山火为什么难以控制?》。尽管如此, 应用元胞自动机模拟传染病相比于 SIR 和 SEIR 模型也有自身优点, 例如可以考虑传染病空间上的传播特点。现在武汉不仅封城, 城内的公共交通也被停掉了, 肺炎的传播只能在染病者周边进行, 这一点非常适合元胞自动机进行模拟。我们参考 SEIR 模型, 构造如图 7 所示的元胞自动机模型。系统中包含四种元胞状态, 分别为易感、潜伏、感染和移除。元胞自动机的规则如下:

- 如果易感元胞 S 的东、南、西、北四个邻居中有潜伏元胞, 则易感元胞下一时刻以概率 P 转变为潜伏元胞 E 。
- 潜伏元胞 E 经过 T 个时间步后转变为感染元胞 I 。
- 感染元胞 I 经过 D 个时间步后转变为移除者 R 。

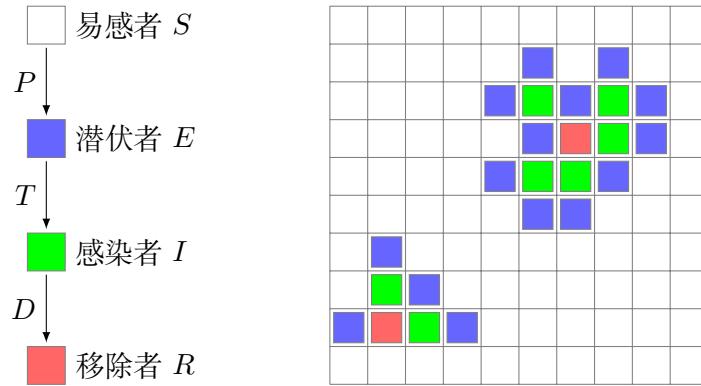


图 7: 传染病元胞自动机模型

5.1 参数

在元胞自动机模型中假设平均潜伏期 $T = 7$ 天, 平均治愈时间 $D = 14$ 天。在 SIR 模型中, 我们估算了 $R_0 = 3.6$, 因此假设每个潜伏元胞在潜伏期内平均能感染 4 个邻居中的 3.6 个, 因此每一个时间步邻居被感染的概率为

$$P = \frac{R_0}{4 \times T} \quad (5.1)$$

实际模拟中, 我们随机地设置了每个元胞的潜伏期和治愈时间, 并分别服从 $N(T, T/2)$ 和 $N(D, D/2)$ 的正态分布。

5.2 结果

我们应用上述元胞自动机模型对 300×300 的空间进行模拟, 初始时刻有一定比例的元胞为潜伏状态。模拟结果如图 8 所示, 所使用的程序见附录代码 6。图 8(b) 所示的四种人群随时间的变化与 SIR 和 SEIR 模型给出的非常相似, 这也证明了元胞自动机确实能够模拟传染病的流行。但仔细比较

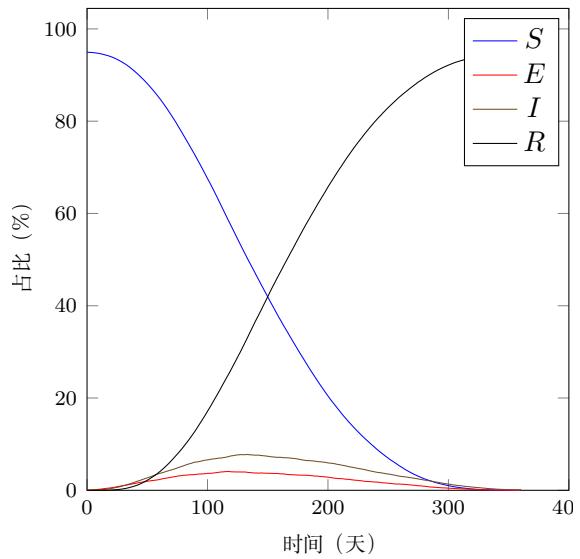


图 8: 元胞自动机模型结果。(a) 元胞空间状态; (b) 四种人群随时间的变化。

元胞自动机与 SIR 和 SEIR 模型的结果还是能发现明显的差别。此外, 元胞自动机的参数很难确定, 因此大模头不推荐用元胞自动机研究大规模的传染病问题。

6 结论

- 回溯传播模型结果表明，武汉在 2020 年 1 月 18 日感染人数可能就已经超过 2700 人，报道出来的确诊病例只占实际的 5 %。
- SIR 模型表明，如果不加控制武汉市疫情在 1 月 20 日左右爆发，3 月上旬左右达到高峰，4 月中旬接近尾声，最终死亡人数将超过 32 万人。
- 考虑到目前政府的管控，预计武汉累计感染人数将控制在 25000 人以内，累计死亡人数将控制在 720 人以内。如果政府有进一步的措施，比如研制新药、疫苗等，实际感染和死亡人数可能比模型预测少。
- 根据数据拟合，新型冠状病毒基本传染数约为 3.6，大于非典和埃博拉。

参考文献

- [1] Wikipedia contributors. 2019–20 wuhan coronavirus outbreak — Wikipedia, the free encyclopedia, 2020. https://en.wikipedia.org/wiki/2019-20_Wuhan_coronavirus_outbreak.
- [2] Walter Stanish. 2019 wuhan coronavirus data (2019-ncov), 2020. <https://github.com/globalcitizen/2019-wuhan-coronavirus-data>.
- [3] Natsuko Imai, Ilaria Dorigatti, Anne Cori, Steven Riley, and Neil M Ferguson. Estimating the potential total number of novel coronavirus cases in wuhan city, china. 2020. <https://www.imperial.ac.uk/media/imperial-college/medicine/sph/ide/gida-fellowships/2019-nCoV-outbreak-report-17-01-2020.pdf>.
- [4] Jonathan M Read, Jessica RE Bridgen, Derek AT Cummings, Antonia Ho, and Chris P Jewell. Novel coronavirus 2019-ncov: early estimation of epidemiological parameters and epidemic predictions. *medRxiv*, 2020.
- [5] 吴翔. 2019-ncov 疫情传播模型.
- [6] Xiaohua Yu. Estimating infected population of wuhan coronavirus in different policy scenarios by sir model, 2020. <http://uni-goettingen.de/en/infectious+diseases/619691.html>.
- [7] Wu JT Leung K and Leung GM. Nowcasting and forecasting the wuhan 2019-ncov outbreak, 2020. <http://www.nhc.gov.cn>.
- [8] 人民日报微博. 武汉市市长：约 500 多万人离开了武汉. http://news.china.com.cn/2020-01-26/content_75650784.htm.
- [9] Korea Centers for Disease Control. The first imported case of the novel coronavirus (2019-ncov) in korea, 2020. https://www.cdc.go.kr/board/board.es?mid=a3040200000&bid=0030&act=view&list_no=365797&tag=&nPage=1.
- [10] World Health Organization. Novel coronavirus –thailand (ex-china), 2020. <https://www.who.int/csr/don/14-january-2020-novel-coronavirus-thailand-ex-china/>.
- [11] Ministry of Health Japan. 新型コロナウイルスに感染した肺炎の患者の発生について, 2020. https://www.mhlw.go.jp/stf/newpage_08906.html.
- [12] Isaac I Bogoch, Alexander Watts, Andrea Thomas-Bachli, Carmen Huber, Moritz UG Kraemer, and Kamran Khan. Pneumonia of unknown etiology in wuhan, china: Potential for international spread via commercial air travel. *Journal of Travel Medicine*, 2020.
- [13] MedSci. 新型冠状病毒感染肺炎疫情实时动态. <http://m.medsci.cn/wh.asp>.
- [14] 萨拉力曼. 根据目前数据，再次分析武汉新冠肺炎患者数量和死亡率. <https://user.guancha.cn/main/content?id=232461>.
- [15] Wikipedia contributors. Basic reproduction number — Wikipedia, the free encyclopedia, 2020. https://en.wikipedia.org/wiki/Basic_reproduction_number.
- [16] 新京报. 武汉市长承认信息披露不及时：获授权后我才能披露. <http://news.sina.com.cn/c/2020-01-27/doc-ihnzha4927544.shtml>.

附录

A SIR 模型

代码 1: 感染率拟合程序 fitb.m

```

1 %% 微信公众号: 数学模型 (MATHmodels)
2 % 联系方式: mathmodels@outlook.com
3
4 % 以 2019 年 12 月 8 日为第 0 天, 起始感染者为 1 人
5 %% 政府官方公布的数据
6 t = [0, 42, 43, 44, 45, 46]';           % 时间
7 I = [1, 198, 218, 320, 478, 639]';         % 感染人数
8
9 %% 回溯传播模型估算的数据
10 % t = [0, 36, 42]';                      % 时间
11 % I = [1, 1182, 2758]';                   % 感染人数
12
13 ft = fittype('exp((5*b-1/14)*x)'); % k = 5; D = 14;
14 f = fit(t,I,ft, 'Startpoint', 0.1)
15 plot(f,t,I, 'o')

```

代码 2: 微分方程组函数 odesir.m

```

1 %% 微信公众号: 数学模型 (MATHmodels)
2 % 联系方式: mathmodels@outlook.com
3
4 function dy = odesir(t, y, beta, gamma, N)
5 dy = [-beta*y(1)*y(2)/N;
6        beta*y(1)*y(2)/N - gamma*y(2);
7        gamma*y(2)];

```

代码 3: SIR 模型求解程序 sir.m

```

1 %% 微信公众号: 数学模型 (MATHmodels)
2 % 联系方式: mathmodels@outlook.com
3
4 N = 11000000;                         % 武汉人口
5 ts = [0, 270];                          % 求解时间区间
6 y0 = [N-1, 1, 0];                       % 初始条件 [S0 I0 R0]
7 C = 14;                                % 感染的平均持续时间
8 gamma = 1/C;                            % 感染者每天平均接触人数
9 k = 5;                                  % 接触时的传染概率
10 b = 0.04133;                           % 接触时的传染概率
11 % b = 0.05214;
12 beta = k*b;                            % 死亡率
13 d = 0.03;                               % 死亡率
14
15 [t, y] = ode45(@odesir, ts,y0, [], beta, gamma, N);
16 y(C+1:end,4) = (N-y(1:end-C,1))*d;
17 plot(t,y);
18 xlabel('Time (days)'); ylabel('Population')
19 legend('S', 'I', 'R', 'D')

```

代码 4: 不同管控强度 SIR 模型求解程序 sirgov.m

```

1 %% 微信公众号: 数学模型 (MATHmodels)
2 % 联系方式: mathmodels@outlook.com

```

```

3
4 N = 11000000; % 武汉人口
5 C = 14; % 感染的平均持续时间
6 gamma = 1/C;
7 % b = 0.04133;
8 b = 0.05214; % 接触时的传染概率
9 d = 0.03; % 死亡率
10
11 % 第一阶段：政府未管控
12 ts = [0, 46];
13 y0 = [N-1, 1, 0];
14 k = 5;
15 beta = k*b;
16 [t1, y1] = ode45(@odesir, ts, y0, [], beta, gamma, N);
17
18 % 第二阶段：政府管控
19 y0 = y1(end,:);
20 ts = [47, 250];
21 k = 1.0;
22 beta = k*b;
23 [t2, y2] = ode45(@odesir, ts, y0, [], beta, gamma, N);
24
25 % 合并两个阶段
26 t = [t1; t2(2:end)];
27 y = [y1; y2(2:end,:)];
28 y(C+1:end,4) = (N-y(1:end-C,1))*d;
29 plot(t,y(:,2:end));
30 xlabel('Time (days)'); ylabel('Population')
31 legend('I', 'R', 'D')

```

B SEIR 模型

代码 5: SEIR 模型求解程序 sier.m

```

1 %% 微信公众号：数学模型 (MATHmodels)
2 % 联系方式：mathmodels@outlook.com
3
4 N = 11000000; % 武汉人口
5 ts = [0, 250]; % 求解时间区间
6 y0 = [N-1, 1, 0, 0]; % 初始条件 [S0 I0 R0]
7 C = 14; % 感染的平均持续时间
8 gamma = 1/C;
9 k = 5; % 感染者每天平均接触人数
10 b = 0.05214; % 接触时的传染概率
11 beta = k*b;
12 alpha = 1/7;
13 [t, y] = ode45(@odeseir, ts, y0, [], beta, gamma, alpha, N);
14 plot(t,y);
15 xlabel('Time (days)'); ylabel('Population')
16 legend('S', 'E', 'I', 'R')
17
18 %% -----
19
20 function dy = odeseir(t, y, beta, gamma, alpha, N)
21 dy = [-beta*y(1)*y(2)/N;
22         beta*y(1)*y(2)/N - alpha*y(2);

```

```

23         alpha*y(2)      - gamma*y(3)
24                           gamma*y(3)];
25 end

```

C 元胞自动机模型

代码 6: 元胞自动机模型程序 ca.m

```

1 %% 微信公众号: 数学模型 (MATHmodels)
2 % 联系方式: mathmodels@outlook.com
3
4 m = 500; n = 500;          % 元胞自动空间大小
5 % 用 1, 2, 3, 4 分别表示 S, E, I, R. 无人区域用 0 表示
6 [S, E, I, R] = deal(1,2,3,4);
7
8 rhoS = 0.95;             % 初始易感人群密度
9 rhoE = 2758/11000000;   % 初始潜伏人群密度
10
11 % X 为每个元胞的状态
12 X = zeros(m,n); X(rand(m,n)<rhoS) = S; X(rand(m,n)<rhoE) = E;
13
14 time = zeros(m,n);      % 计时: 用于计算潜伏时间和治疗时间
15 % 邻居方位 d
16 d = {[1,0], [0,1], [-1,0], [0,-1]};
17
18 T = 7;                  % 平均潜伏期
19 D = 14;                  % 平均治愈时间
20 P = 3.6/T/4;            % R0 = 3.6, 潜伏期平均感染 3.6 个
21
22 % 每个元胞的潜伏期和治愈时间服从均值为T和D的正态分布
23 Tmn = normrnd(T,T/2,m,n); Dmn = normrnd(D,D/2,m,n);
24
25 figure('position',[50,50,1200,400])
26 subplot(1,2,1)
27 h1 = imagesc(X);
28 colormap(jet(5))
29 labels = {'无人', '易感', '潜伏', '发病', '移除'};
30 lcolorbar(labels);
31 subplot(1,2,2)
32 h2 = plot(0, [0,0,0,0]); axis([0,400,0,m*n])
33
34 for t = 1:450
35     % 邻居中潜伏和发病的元胞数量
36     N = zeros(size(X));
37     for j = 1:length(d)
38         N = N + (circshift(X,d{j})==E|circshift(X,d{j})==I);
39     end
40
41     % 分别找出四种状态的元胞
42     isS = (X==S); isE = (X==E); isI = (X==I); isR = (X==R);
43
44     % 将四种状态的元胞数量存到 Y 中
45     Y(t,:) = sum([isS(:) isE(:) isI(:) isR(:)]);
46
47     % 计算已经潜伏的时间和已经治疗的时间
48     time(isE|isI) = time(isE|isI) + 1;

```

```
49
50 % 规则一：如果 S 邻居有 N 个染病的，则 S 以概率 N*P 变为 E，否则保持为 S
51 ifS2E = rand(m,n)<(N*P);
52 Rule1 = E*(isS & ifS2E) + S*(isS & ~ifS2E);
53
54 % 规则二：如果 E 达到潜伏期，则转变为 I，否则保持为 E
55 ifE2I = time>Tmn;
56 Rule2 = I*(isE & ifE2I) + E*(isE & ~ifE2I);
57 time(isE & ifE2I) = 0;
58
59 % 规则三：如果 I 达到治愈时间，则转变为 R，否则保持为 I
60 ifI2R = time>Dmn;
61 Rule3 = I*(isI & ~ifI2R) + R*(isI & ifI2R);
62
63 % 规则四：已经治愈 R 有抗体，保持为 R
64 Rule4 = R*isR;
65
66 % 叠加所有规则，更新所有元胞状态
67 X = Rule1 + Rule2 + Rule3 + Rule4;
68
69 set(h1, 'CData', X);
70 for i = 1:4; set(h2(i), 'XData', 1:t, 'YData', Y(1:t,i)); end
71 drawnow
72 end
```