

# 基于SIR的计算机病毒时滞传播模型 (M.S.I.R.)

黎锦灏 518021910771

## 摘要

在计算机网络中, 病毒感染计算机节点后往往会在网络中传播, 因病毒感染而存在的资源暴露风险需要我们研究病毒进入网络后传播的性质特点, 相应提出病毒防范策略。在原有SIR模型的基础上, 本文引入了传播时滞性的概念, 建立了一个新的传染病SIR模型 (Modified Susceptible-Infection-Recovery, 即M.S.I.R.模型)。模型根据病毒传播发展条件来处理网络中节点的感染、恢复与获得免疫, 讨论了无时延情形的系统稳定性, 最后通过数值模拟和数据分析, 体现了模型在具体数值上的运行结果。

## 关键词

计算机病毒 网络 传染病模型 时延 稳定性

## 一、模型简介与应用

随着网络的快速发展, 将计算机连接成一个庞大的网络是现代数字通信时代网络应用的趋势。计算机系统连接到网络的次数越多, 恶意代码进入计算机并攻击系统、造成损失的可能性就越大, 这对于个人和组织来说是难以避免的。计算机病毒种类繁多, 针对病毒的防范措施往往滞后于病毒的传播。考虑到计算机病毒的爆发与人类传染病的传播的类似性, 许多常用的医学传染病模型值得借鉴。

传统的SIR模型能较准确地描绘传染病的传播情形, 但也存在不足, 例如未考虑传播的时延性。考虑计算机病毒感染的情形, 感染病毒的文件并不一定立刻传播病毒给其他文件, 只有在被感染病毒的同时, 处于激活状态并且获得了相应权限, 该病毒才会传播给网络或计算机系统其他文件。我们不难发现, 有时病毒可能在较长一段时间内无法被检测到, 因为被感染的文件并没有被调用, 也没有任何感染其他文件的行为。在这样的背景下, M.S.I.R.模型在原先S.I.R.模型的基础上引入了时延的概念, 并修改了原有的数学模型, 增强了区域稳定性, 具体细节将在之后讨论。

在当今数字通信时代, 无病毒网络难以实现, 因此分析流行病毒的传播模式, 并预测新病毒在群体中的传播情形就显得极为重要, 这一工作能给病毒防范与杀灭提供有力保障。

## 二、数学模型

在原有SIR模型基础上, M.S.I.R.模型修改了部分假设, 并引入了时滞性的概念。

假设节点网络中的每一个节点都视为是一台计算机, 节点有三种可能状态:

易感个体 (S): 当前健康但是容易被病毒感染的个体

感染个体 (I): 已经被感染并且能传播病毒给健康节点的个体

治愈个体 (R): 感染过并已经被治愈, 无法被再次传染或传播病毒的个体

在SIR模型里，病毒最开始传播的时候所有人都是易感者，也就是所有人都有可能感染，当一部分人在接触到病毒之后就变成了感染者，而感染者会接受各种治疗，最后变成了治愈者。

SIR模型有两个基本假设：

- (1)、已被恶意代码感染的文档/计算机具有免疫力
- (2)、恶意代码的潜伏期很短，近似地认为等于0（注：M.S.I.R.模型引进了时延 $\tau$ 的概念）

下面讨论M.S.I.R.的数学模型：

初始条件满足

$$S(t) \geq 0, I(t) \geq 0, R(t) \geq 0, t \in (-\tau, 0)$$

若定义 $b$ 为新增计算机数， $p$ 为免疫计算机比例， $\beta$ 为感染个体的感染比例， $d$ 为计算机的消亡比例， $\delta$ 为治愈个体的消失率， $\alpha$ 为由于病毒造成的计算机减少比例， $\gamma$ 为感染个体的治愈比例， $\sigma$ 为饱和常数，则我们有

$$\begin{aligned}\frac{dS}{dt} &= (1-p)b - \frac{\beta S(t-\tau)I(t-\tau)}{1+\sigma S(t-\tau)} - dS + \delta R \\ \frac{dI}{dt} &= \frac{\beta S(t-\tau)I(t-\tau)}{1+\sigma S(t-\tau)} - (d+\alpha+\gamma)I \\ \frac{dR}{dt} &= \gamma I + pb - (d+\delta)R\end{aligned}$$

下面分析系统在无时延情况下的稳定性，令变化率为0，即群体趋于稳定的情形时有

$$\begin{aligned}(1-p)b - \frac{\beta S(t-\tau)I(t-\tau)}{1+\sigma S(t-\tau)} - dS + \delta R &= 0 \\ \frac{\beta S(t-\tau)I(t-\tau)}{1+\sigma S(t-\tau)} - (d+\alpha+\gamma)I &= 0 \\ \gamma I + pb - (d+\delta)R &= 0\end{aligned}$$

分别解得稳定点的对应条件

$$\begin{aligned}S^* &= \frac{d+\alpha+\gamma}{\beta-\sigma(d+\alpha+\gamma)} \\ I^* &= \frac{(1+\sigma S^*)((1-p)b - dS^* + \delta R^*)}{\beta S^*} \\ R^* &= \frac{\gamma I^* + pb}{d+\delta}\end{aligned}$$

要使得稳定点落在正区间（即以上三个坐标均位于正半轴），则必须满足

$$\begin{aligned}\sigma(d+\alpha+\gamma) &< \beta \\ S^* &= \frac{(1-p)b + \delta R^*}{d}\end{aligned}$$

而无时延情形若想满足稳定性还要求

$$\delta(1+\sigma S^*)^2 + \beta I^* > 0,$$

$$\beta S^* + (\alpha + d + \gamma)(1 + \sigma S^*) > 0,$$

$$d + \gamma > 0$$

因此，对于E(S,I,R)而言是趋于局部稳定的。

### 三、数值仿真与数据分析

我自己用python编写代码求解模型的三个微分方程，并假设初始值

$$S_0 = 3.6, I_0 = 1.5, R_0 = 1.2$$

为了便于数据模拟，我选取了如下参数列表：

p	b	$\beta$	d	$\alpha$	$\gamma$	$\sigma$	$\delta$
0.9	1	0.85	0.3	0.3	0.3	0.4	0.7

由此可作出S、I、R随时间变化的图像。

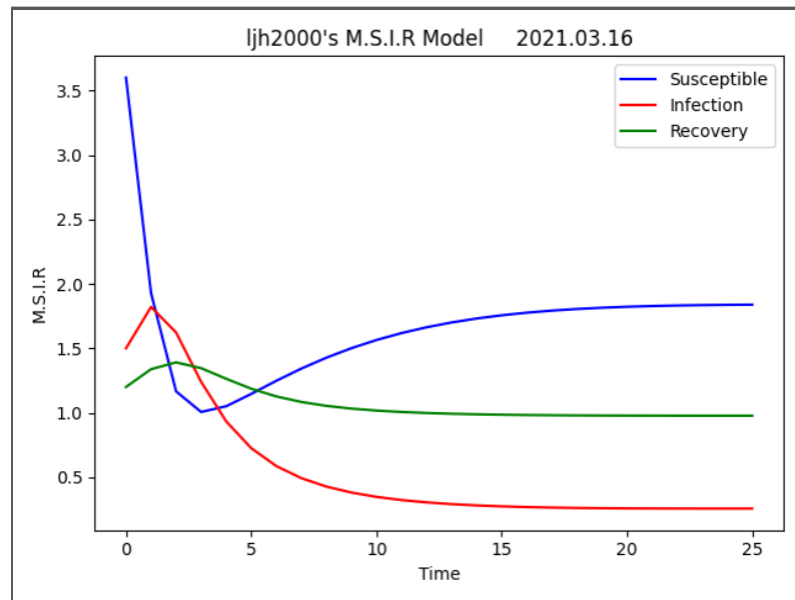


图1. S(t)、I(t)、R(t)的时间序列

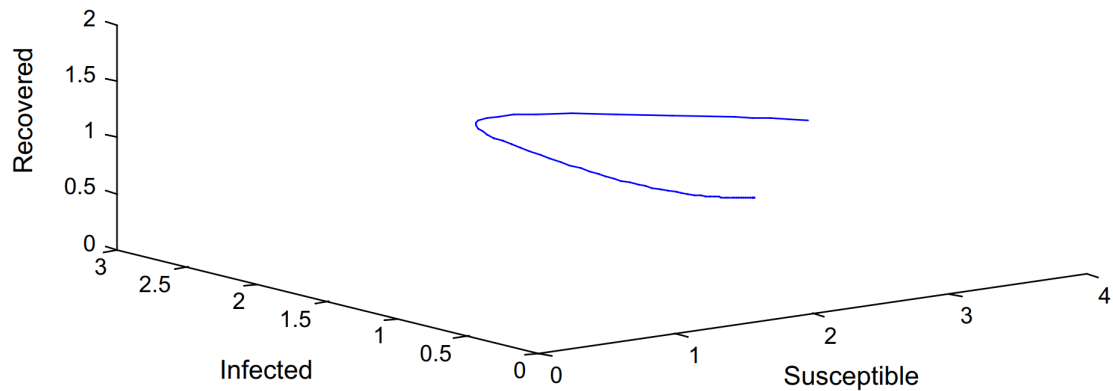


图2. S、I、R的相位图

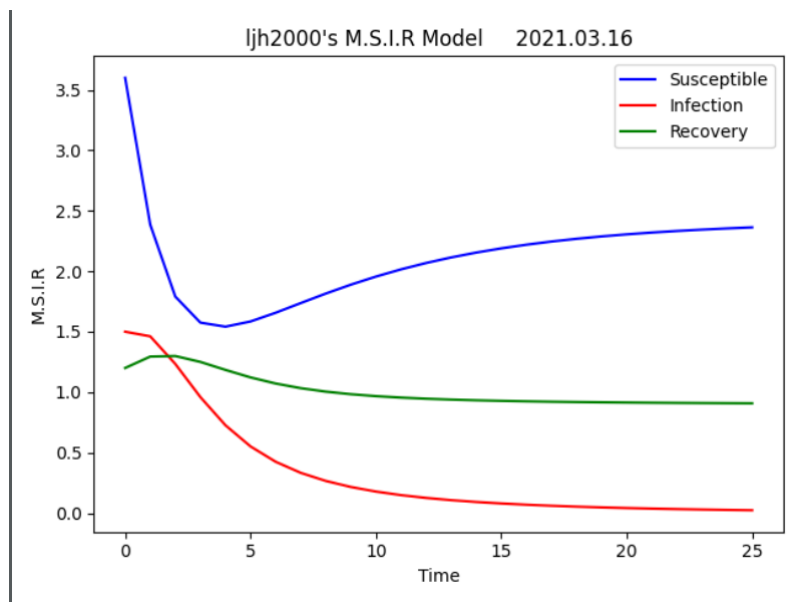


图3.  $\beta=0.65$ 时,  $S(t)$ 、 $I(t)$ 、 $R(t)$ 的时间序列

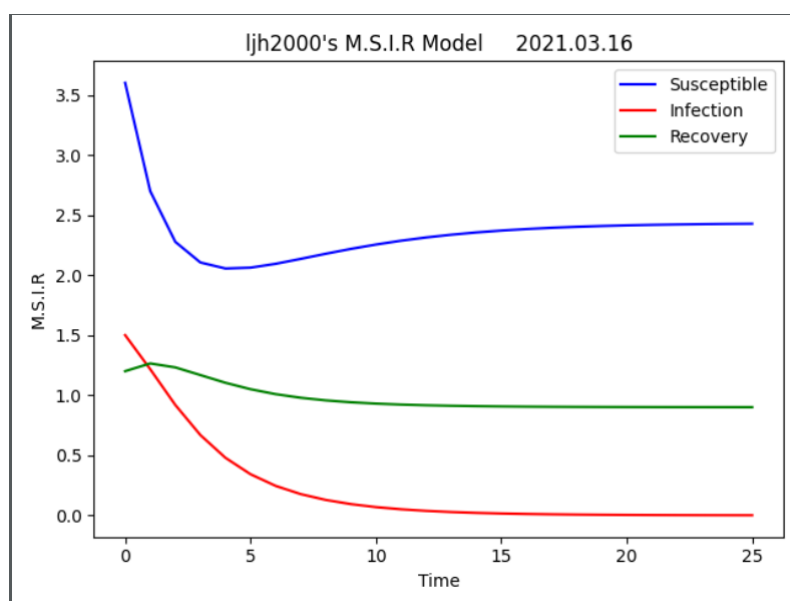


图4.  $\beta=0.5$ 时,  $S(t)$ 、 $I(t)$ 、 $R(t)$ 的时间序列

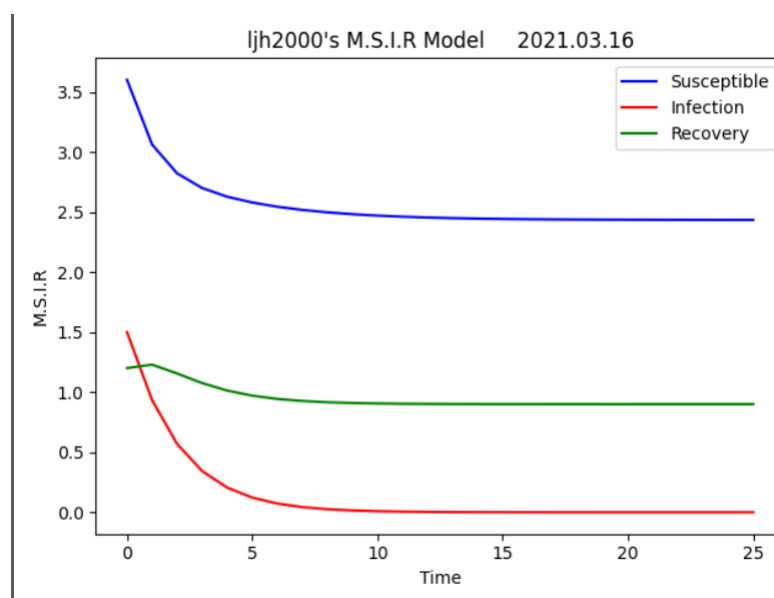


图5.  $\beta=0.3$ 时,  $S(t)$ 、 $I(t)$ 、 $R(t)$ 的时间序列

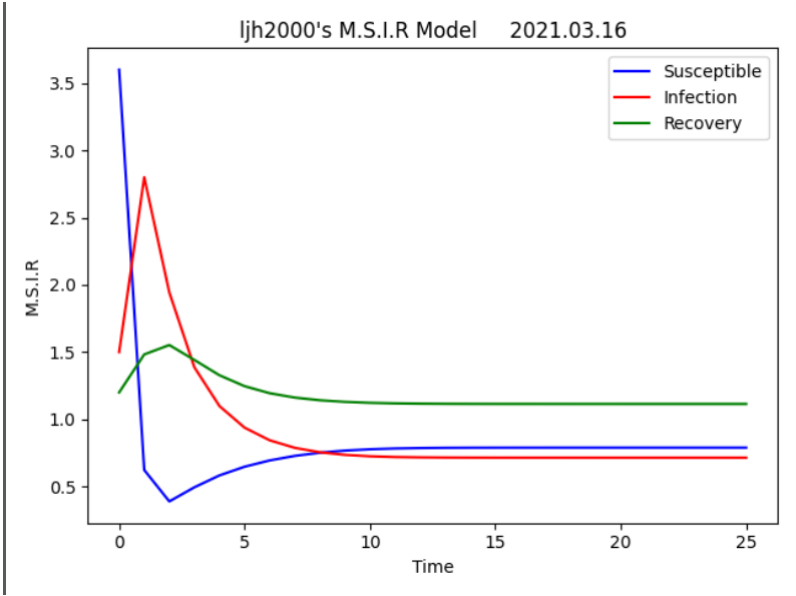


图6.  $\beta=1.5$ 时,  $S(t)$ 、 $I(t)$ 、 $R(t)$ 的时间序列

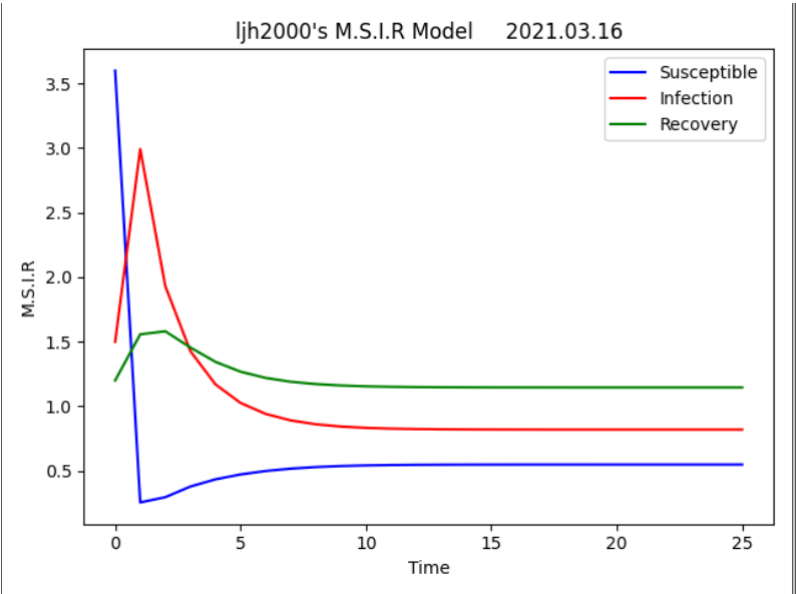


图7.  $\beta=2.0$ 时,  $S(t)$ 、 $I(t)$ 、 $R(t)$ 的时间序列

从图中我们可以观察到,

- 1、若 $\beta < 1$ , 随着 $\beta$ 的逐渐减少, 易感群体和治愈群体比例逐渐增大, 感染群体的比重逐渐减小, 并且感染群体在某个时刻后消失 (对比图3、4、5) ;
- 2、若 $\beta > 1$ 且逐渐增加, 治愈群体比重会稳定上升, 感染群体也会缓慢增加, 而易感群体的比重持续减小 (对比图6和7) 。
- 3、若令治愈率随感染率同步变化, 则感染群体先下降而后缓慢上升直至趋于稳定。这一上升趋势反映了被感染的个体并没有立刻被治愈从而转变为治愈个体, 所以感染群体始终在系统里占据一定比重。

## 四、结论

为了建立疾病传播与计算机病毒的联系，本模型在SIR模型的基础上引进了时延与恢复免疫，在对于时延进行灵敏度分析后得到了平衡点的局部稳定性。通过对易感群体、感染群体、免疫群体时间序列的数值模拟和数据分析，得到了感染率、恢复速率和群体变换趋势的相关规律。

## 五、参考文献

1. MadhuSudanan, V., Geetha, R. Dynamics of Epidemic Computer Virus Spreading Model with Delays. Wireless Pers Commun 115, 2047–2061 (2020). <https://doi.org/10.1007/s11277-020-07668-6>
2. Gan, C., Feng, Q., Zhu, Q. et al. Analysis of computer virus propagation behaviors over complex networks: a case study of Oregon routing network. Nonlinear Dyn 100, 1725–1740 (2020). <https://doi.org/10.1007/s11071-020-05562-1>
3. 刘功申. 计算机病毒与恶意代码[M]. 北京：清华大学出版社，2021.

## 六、附录

1、python程序代码：

```
#M.S.I.R Model 黎锦灏 518021910771
import scipy.integrate as spi
import numpy as np
import matplotlib.pyplot as plt

# 传播时间
T = 25

#参数初值
p = 0.9
b = 1
beta = 1.5
d = 0.3
alpha = 0.3
gamma = 0.3
sigma = 0.4
delta = 0.7

#S.I.R.
S0 = 3.6
I0 = 1.5
R0 = 1.2

# Initial Condition
INI = (S0,I0,R0)
```

```

def funcSIR(inivalue,_):
    Y = np.zeros(3)
    X = inivalue
    # Susceptible 易感个体变化
    Y[0] = (1 - p) * b - beta * X[0] * X[1] / (1 + sigma * X[0]) - d * X[0]
+delta * X[2]
    # Infection 感染个体变化
    Y[1] = beta * X[0] * X[1] / (1 + sigma * X[0]) - (d + alpha + gamma) * X[1]
    # Recovery 治愈个体变化
    Y[2] = gamma * X[1] + p * b - (d + delta) * X[2]
    return Y

T_range = np.arange(0,T + 1)

#求解微分方程并作图
RES = spi.odeint(funcSIR,INI,T_range)

plt.plot(RES[:,0],color = 'blue',label = 'Susceptible',marker = '')
plt.plot(RES[:,1],color = 'red',label = 'Infection',marker = '')
plt.plot(RES[:,2],color = 'green',label = 'Recovery',marker = '')
plt.title('1jh2000\'s M.S.I.R Model 2021.03.16')
plt.legend()
plt.xlabel('Time')
plt.ylabel('M.S.I.R')
plt.show()

```