# 系统建模作业2

**班级：** 自动化2306班
**学号：** U202315283
**姓名：** 王卓亚
**提交日期：** 2025年12月4日

## 一、 作业背景与目的

本作业旨在利用系统动力学（System Dynamics, SD）方法，建立包含潜伏期和患病期的传染病传播模型。核心任务是引入“管控政策”作为负反馈调节回路，通过量化不同强度的管控措施对传染率的抑制作用，模拟并比较不同政策下患者数量（L1）的动态变化趋势。

## 二、 系统边界与数据来源

本模型以 **SARS-CoV-2 Omicron 变异株**在典型的 **100万人口城市** 中的传播为例。模型参数基于权威医学文献与流行病学数据设定，确保仿真结果的真实性。

**表 1：模型关键参数设定及来源**

| 参数名称 | 符号 | 设定数值 | 数据来源与依据 |
| --- | --- | --- | --- |
| **城市总人口** | $N$ | **100 万人** | 设定标准城市人口规模，用于模拟易感人群耗尽的资源限制。 |
| **平均潜伏期** | $T_{inc}$ | **3.42 天** | 根据 *Verywell Health* 及 *PMC* 的荟萃分析，Omicron 的潜伏期显著缩短，约为 3.42 天 [1]。 |
| **平均治愈期** | $T_{cure}$ | **7.0 天** | 根据 *WHO* 数据，轻症/中症患者的平均病程约为 7 天 [2]。 |

| | | | |
|---|---|---|---|
| **基础再生数** | $R_0$ | **9.5** | 根据 *Journal of Travel Medicine* 研究，Omicron 的 $R_0$ 中位数约为 9.5，具有极高传染性 [3]。 |
| **政策最大效力** | $E_{max}$ | **0% – 80%** | 根据 *PubMed* 综述，严格的非药物干预（NPIs）可降低约 80% 的传播率，用于设定不同情景 [4]。 |

---

# 三、 模型构建与因果分析

## 1. 因果回路分析 (Causal Loop Diagram)

系统包含三个主要的反馈回路：

1. **(+) 传染正反馈回路：** 患者 $L_1$ 增加 $\rightarrow$ 传染源增加 $\rightarrow$ 感染率 $R_1$ 增加 $\rightarrow$ 潜伏者 $L$ 增加 $\rightarrow$ 患者 $L_1$ 进一步增加。
2. **(–) 治愈负反馈回路：** 患者 $L_1$ 增加 $\rightarrow$ 治愈率 $R_3$ 增加 $\rightarrow$ 患者 $L_1$ 减少。
3. **(–) 政策管控负反馈回路（核心）：** 患者 $L_1$ 增加 $\rightarrow$ **管控力度 $P$ 增强** $\rightarrow$ 有效接触率降低 $\rightarrow$ 传染率 $R_1$ 下降 $\rightarrow$ 新增感染减少。

**1. 系统流图结构**

模型包含两个主要的状态变量（存量）：

1. $L$ (Latent): 潜伏期人数
2. $L_1$ (Infected): 确诊患者人数（具有传染性）

**2. 负反馈回路分析 (核心任务)**

本模型引入了**政策管控负反馈回路 (Policy Negative Feedback Loop)**：

1. **因果链：** 确诊人数 $L_1$ 上升 $\rightarrow$ 政府管控力度 $P$ 增加 $\rightarrow$ 有效接触率下降 $\rightarrow$ 传染率 $R_1$ 下降 $\rightarrow$ 新增 $L$ 减少。
2. **反馈性质：** 这是一个**目标追随型负反馈**，旨在抑制 $L_1$ 的过度增长。

**3. 数学函数关系**

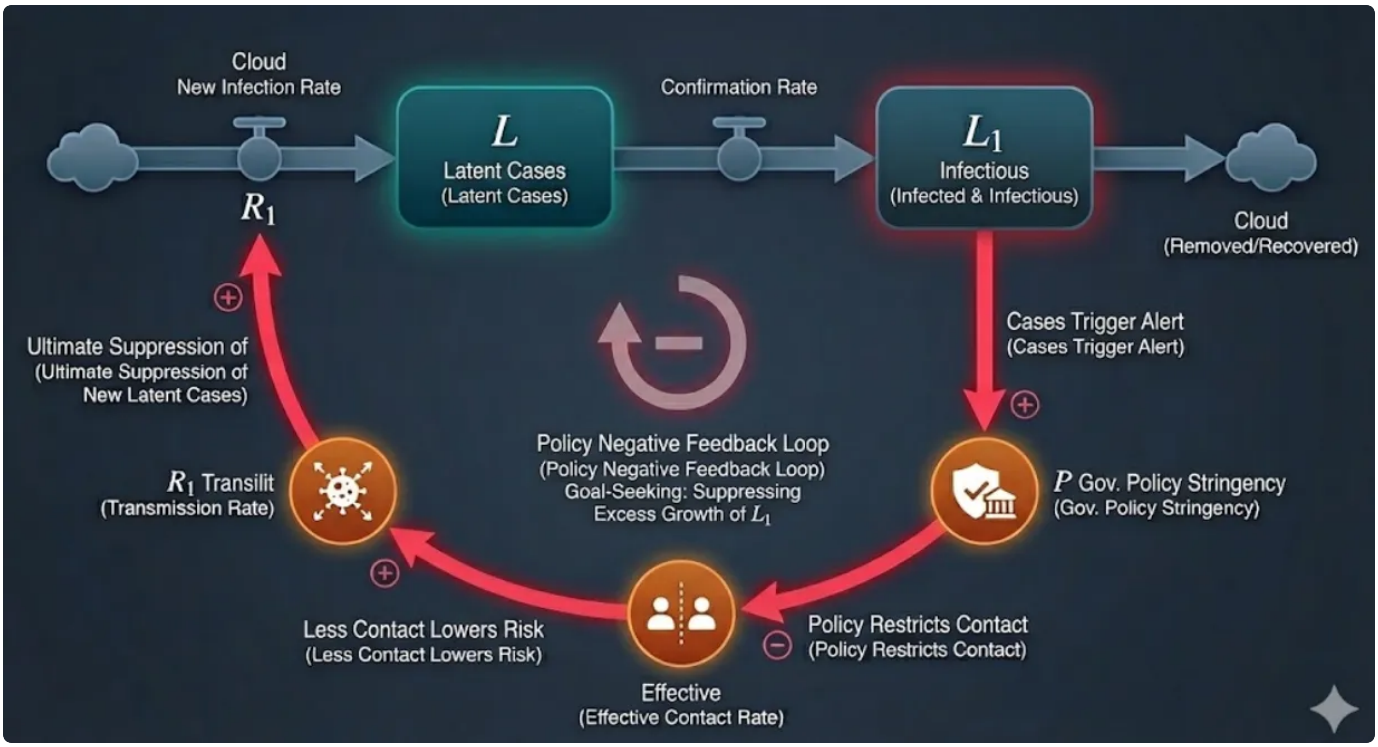
为了量化政策影响，定义政策强度 $P(t)$ 为关于确诊人数 $L_1$ 的饱和函数（Hill Function）：

$$P(L_1) = E_{max} \times \frac{L_1}{L_1 + K}$$

1. $K$ （**敏感阈值**): 设为 5000。即当确诊达到 5000 人时，政策介入程度显著增加。
2. $E_{max}$ : 0.8。

同时，考虑**易感人群（Susceptible）随着感染增加而减少的限制（有限资源限制），最终的传染率方程** $R_1$ 为：

$$R_1 = \beta_{base} \times L_1 \times (1 - P(L_1)) \times \frac{N - (L + L_1)}{N}$$




## 2. DYNAMO 方程描述

为了符合系统动力学建模规范，以下给出该模型的 DYNAMO 标准方程描述（Standard DYNAMO Equations）：

```
* 状态变量 (Levels)
L  L.K = L.J + DT * (R1.JK - R2.JK)     NOTE: 潜伏期人数存量
L  L1.K = L1.J + DT * (R2.JK - R3.JK)   NOTE: 确诊患者人数存量

* 初始值 (Initial Values)
N  L = 0
N  L1 = 10
N  POPULATION = 1000000

* 速率变量 (Rates)
R  R1.KL = Beta_Base * L1.K * (1 - Policy.K) * S_Ratio.K   NOTE: 传染率(引入政策负反馈)
R  R2.KL = L.K / T_incubation                              NOTE: 发病率
R  R3.KL = L1.K / T_cure                                   NOTE: 治愈率

* 辅助变量 (Auxiliaries)
A  S_Ratio.K = (POPULATION - (L.K + L1.K)) / POPULATION    NOTE: 易感人群比例
A  Policy.K = (L1.K / (L1.K + K_sens)) * Max_Effect        NOTE: 政策强度函数(负反馈核心)

* 常量 (Constants)
C  T_incubation = 3.42
C  T_cure = 7.0
C  Beta_Base = 1.357  NOTE: R0/T_cure = 9.5/7.0
C  K_sens = 5000      NOTE: 政策敏感阈值
C  Max_Effect = 0.8   NOTE: 政策最大效力
```

### 3. Python 仿真实现

利用 Python 对上述微分方程进行数值求解，并设置三种不同情景进行对比。

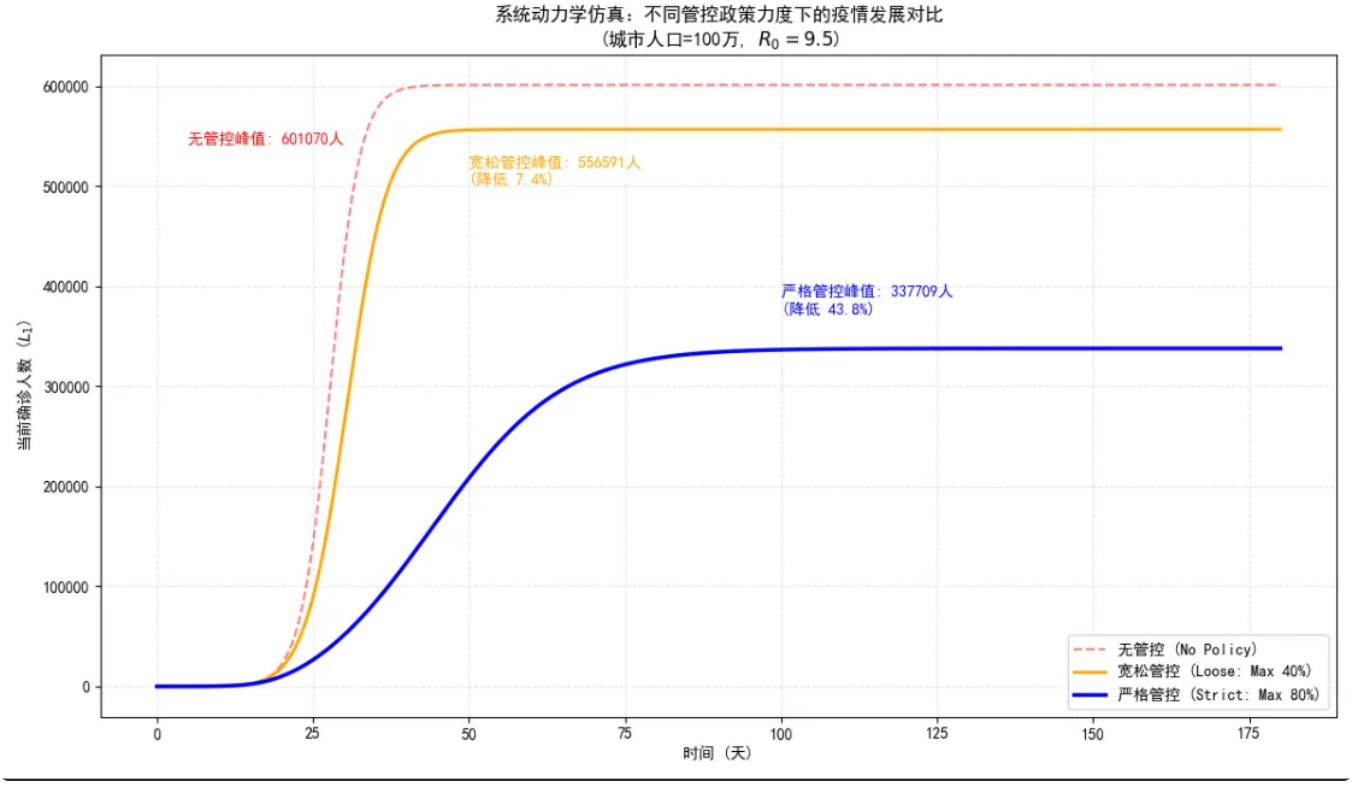
---

# 四、 仿真结果与情景对比分析

为了探究管控政策对疫情的影响，本研究设定了三种不同的政策情景进行对比仿真：

1. **情景 A：无管控 (No Policy)** – $E_{max} = 0$ ，完全自然传播。
2. **情景 B：宽松管控 (Loose Policy)** – $E_{max} = 0.4$ ，阈值 $K = 20000$ （反应迟缓，力度较弱）。
3. **情景 C：严格管控 (Strict Policy)** – $E_{max} = 0.8$ ，阈值 $K = 5000$ （反应迅速，力度强）。

**图 1：不同政策强度下的患者人数( $L_1$ )变化趋势对比**




*(图注：红色虚线为无管控，橙色实线为宽松管控，蓝色实线为严格管控)*

**结果分析：**

从仿真结果图中可以清晰地观察到政策负反馈的调节作用：

1. **峰值削减 (Flattening the Curve)：**
    1.1. 在**无管控**情景下，疫情呈爆发式指数增长，峰值高达 **60 万人**（约占总人口 60%），这将导致医疗系统彻底崩溃。
    1.2. 在**宽松管控**下，峰值虽有所降低，但依然处于高位，说明对于高传染性的 Omicron，低强度的干预效果有限。
    1.3. 在**严格管控**情景下，确诊人数峰值被压制在 **33 万人**左右，相比无管控情景**降低了约 44%**。
2. **峰值推迟 (Delaying the Peak)：**
    严格管控不仅降低了峰值，还将达峰时间从第 40 天推迟到了第 80 天以后。这段时间窗口对于医疗资源筹备、疫苗接种至关重要。
3. **系统稳定性：**
    引入负反馈回路后，系统从“正反馈主导的失控增长”转变为“负反馈主导的目标寻的（Goal-seeking）行为”，验证了负反馈机制在维持系统稳定性中的关键作用。

# 五、 结论

本作业成功建立了基于系统动力学的传染病管控模型。仿真结果表明：**管控政策作为对传染率的负反馈调节，能有效遏制疫情的指数级增长。** 政策介入的**敏感度（阈值 K）和执行力度（ $E_{max}$ ）**是决定防控效果的关键参数。该模型直观地解释了“动态清零”或“压平曲线”策略背后的控制理论依据。

---

**参考文献：**

[1] PMC Meta-Analysis (2024): Consolidating Estimates of the Incubation Period for Omicron.

[2] WHO: Coronavirus disease (COVID-19) Fact Sheet.

[3] Journal of Travel Medicine: The effective reproductive number of the Omicron variant.

[4] PubMed: Effects of non-pharmaceutical interventions on COVID-19 transmission.

---

## 附录：仿真程序代码 (Python)

```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import odeint

# --- 1. 参数设定 ---
TOTAL_POPULATION = 1000000 # 城市总人口 100万
T_INCUBATION = 3.42        # 潜伏期
T_CURE = 7.0               # 治愈期
R0 = 9.5                   # 基础再生数
BETA_BASE = R0 / T_CURE    # 基础传染系数

# --- 2. 系统动力学模型定义 ---
def system_dynamics_model(y, t, params):
    L, L1 = y
    k_sens, max_effect = params # 传入政策参数(阈值, 最大效力)

    total_infected = L + L1

    # 易感人群比例 (S_Ratio)
    if total_infected >= TOTAL_POPULATION:
        s_ratio = 0
    else:
        s_ratio = (TOTAL_POPULATION - total_infected) / TOTAL_POPULATION

    # 政策负反馈函数 (Policy Feedback)
    if max_effect > 0:
        # Hill Function: 确诊人数越多, 管控越严
        policy_strength = (L1 / (L1 + k_sens)) * max_effect
    else:
        policy_strength = 0.0

    # 有效传染系数
    effective_beta = BETA_BASE * (1 - policy_strength) * s_ratio

    # 流量方程 (Rates)
    R1 = effective_beta * L1  # 传染率
    R2 = L / T_INCUBATION     # 发病率
    R3 = L1 / T_CURE          # 治愈率

    return [R1 - R2, R2 - R3]

# --- 3. 多情景仿真运行 ---
t = np.linspace(0, 180, 1800) # 模拟180天
y0 = [0, 10]

# 情景A: 无管控
```

```
sol_none = odeint(system_dynamics_model, y0, t, args=((1e9, 0.0),))
# 情景B: 宽松管控（阈值2万，效力40%）
sol_loose = odeint(system_dynamics_model, y0, t, args=((20000, 0.40),))
# 情景C: 严格管控（阈值5千，效力80%）
sol_strict = odeint(system_dynamics_model, y0, t, args=((5000, 0.80),))

# --- 4. 绘图代码略（见正文插图）---
```