

基于时滞动力学系统的“检测期望-SIJ”循环模型

摘要

为了获得多轮核酸检测的最佳混检率 K , 该文引入了三个数学模型——时滞模型、混检期望模型和 SIJ 传染病模型。

针对问题一, 基于新冠病毒潜伏期长的特点, 该文引入时滞假设, 从而将求未来阳性检出率 s 转化为了求目前的感染率 p : 即

$$s(t) = \int p(t - \tau') G(\tau', \tau) d\tau'$$

, 其中 G 为时间间隔概率分布函数; 接着利用最小二乘法拟合数据, 可获得现在的感染率, 由此得到阳性检出率; 利用阳性检出率, 以每人进行核酸检测的平均次数为目标函数, 就可用数学期望的方法得到最佳混检人数 K 。

针对问题二, 该文以上海和广州为例, 利用问题一中的模型, 分别计算得出两所城市的最佳混检人数分别为 9 人和 21 人。

针对问题三, **目标上**, 基于“疫情期间每次核酸复筛完成后立刻进行下一次核酸”的假设, 该文将目标从“单次检测次数最少”调整为疫情持续时间最短, 从而与一般的传染病预测模型建立起联系。**方法上**, 为了避免传统时滞动力学系统 DDE 方程过于复杂的问题, 该文提出了“时滞更新-检测期望-SIJ 模型-时滞更新”的循环结构: **首先**, 利用核酸检测节点将整个疫情分为一个个周期; **然后, 在同一个周期内**, SIJ 模型不仅能模拟疫情的发展, 同时作为 ODE, 其解函数在周期末结点的值(感染人数比)可用来求解下个阶段的最佳混检率 K ; **在两个周期之间**, 则用时滞模型进行 SIJ 参数的更新。

模型中, I 是未封控人群的比例, J 是已封控人群比例。我们定义转移率 α 是单位时间由 I 转化为 J 的比例, 通过建立 α 与 K 的数学关系, 使得混检率对疫情走势也产生影响。

为了解决单周期最优解未必是全局最优解和穷举法下计算效率过低的问题, 我们引入“**束搜索**”的概念, 使得模型具有一定的“记忆性”, 从而能在多个阶段的可行解中挑选最优解。

然后, 我们将上海和广州本轮的疫情数据代入, 实践证明在通过最小二乘法求最优解后, 时滞模型与现实数据吻合地很好。基于拟合得到的传染率和治愈率, 我们得到了本轮疫情每一轮核酸检测的最佳混检人数 K 。文章发现, 转移率 α 在疫情走势中起到至关重要的作用。

最后, 基于对结果的讨论, 我们对政府提出了一些建议。

1 问题背景与重述

1.1 问题背景

2019年底，在湖北省武汉市华南海鲜市场周围发现了一种感染性极强新型冠状病毒，又正值春运期间，大量人口流动加速了疫情的蔓延，短短几个星期这种病毒就在全国四散开来，随后又在全世界蔓延开来，2020年3月11日，世卫组织认为当前新冠肺炎疫情可被称为全球大流行。新冠疫情对全球造成了极大的影响，并危害着公众的生命安全，截至北京时间2022年4月29日23时47分，全球累计新冠肺炎确诊病例超5亿例，累计死亡病例超600万例。

开展新冠病毒核酸检测，对早诊早治、疫情防控和生产生活都具有重要意义。在密切接触者及公众等人群中进行核酸检测，有助于及早发现感染者，特别是及早发现那些已经感染了病毒但尚未出现症状的人，从而及早采取隔离和治疗措施，既可以避免传染他人又可以减少自身发展成重症的风险。

2022年3月以来，上海又迎来了一轮疫情，本轮新冠肺炎疫情是由奥密克戎变异株引起。该变异株与之前的新冠病毒流行株相比，具有传播力更强，传播速度非常快，而且无症状感染者比例更高，传播隐匿，早期发现更难等特点。核酸检测仍然是帮助上海打赢疫情防控攻坚战的重要武器，而上海作为人口密度高的大城市，核酸检测的效率直接影响到了疫情的防控成果，如何在最短时间内实现居民大范围核酸检测工作是亟待解决的关键问题。

1.2 问题重述

根据题中所给信息，本文将问题细化为以下五个问题，并建立数学模型进行分析和研究。

问题一：为了寻找合理的阳性检出率，建立描述核酸检测阳性人数占比的数学模型，分设若干参数描述其变化规律，确定出合理的阳性检出率。

问题二：在假定阳性检出率已知的基础之上，为了确定每组混检的人数使单轮核酸检测的效率最高，计算出在不同混检人数下一组中每个人检测次数的期望值，寻找适当的混检人数使得该期望值最小。

问题三：为了给出不同地区相应的最佳混检方案，以收集到的不同地区的数据建立数学模型，利用上一个问题的结论给出相应的最佳混检方案。

问题四：在需要多轮检测的情况下，为使疫情尽快结束，建立传染病模型求局部最优解，即给出每一轮的混检方案使得每一轮核酸检测的次数最少。

问题五：在需要多轮检测的情况下，为使疫情尽快结束，建立可动态调节参数的传染病模型求全局最优解，即给出每一轮的混检方案使得多轮检测的总检测次数最少。

2 符号说明

符号	意义
N	地区总人数
K_i	第 i 轮核酸检测的混检人数，即 K_i 人一管
α_i	第 i 轮的转移率
β_i	第 i 轮核酸检测的平均阳性检出率
γ_i	第 i 轮核酸检测的平均阴性检出率，即 $\gamma_i = 1 - \beta_i$
S_i	第 i 轮核酸检测开始前易感人群比例
I_i	第 i 轮核酸检测开始前未封控人群比例
J_i	第 i 轮核酸检测开始前已封控人群比例
t_i	第 i 轮核酸检测需要花费的天数
τ	从感病到被检测出来是阳性的时间间隔 (也可以理解为潜伏期)
$k_1(i)$	第 i 轮核酸检测开始前未封控病人的传染率
k_2	已封控病人的传染率
h_1	未封控病人的治愈率
h_2	已封控病人的治愈率
r	核酸检测总轮数
T	进行 r 轮核酸检测需要花费的总天数
v	核酸检测效率，即一天可以检测的总管数

3 模型假设

我们对进行以下总体的假设，剩下的细节上的假设会在本文的相应部分一一说明：

- (H1) 潜伏期平均天数 τ ，地区总人数 N 和核酸检测速率 v 在一定时间内为恒定值。
- (H2) 在每一轮核酸检测内，一旦复筛完成，立刻进行下一轮核酸检测。
- (H3) 复筛一次将花费相当于第一次混检的 c 倍的资源， c 为大于 1 的常数，
- (H4) 如果进行到第 i 轮检测时，感染人数比例小于 0.00001，那么我们认为疫情已经被完全控制住，即无需进行下一轮核酸检测。
- (H5) 已封控病人 J 的感染率为一较小的常数。
- (H6) 转移率是单位时间内由 I 类病人转变为 J 类病人的比例。在每一轮核酸检测内，感病者的转移率保持常数。但在两轮核酸检测之间，感病者的转移率 α_i 与阴性管

数呈正比。

对于 (H2)，如果两轮核酸检测之间相隔时间过长，那么疫情会在这段时间内加重，不利于疫情的控制。因此我们假设两轮核酸检测之间没有时间间隔。

对于 (H6)，在现实中，每两轮核酸检测之间都有时间差。若检测为阴性的管数越多，那么需要进行复筛的时间就会越少，从而导致两轮核酸检测之间的时间差减少，这样会使得在复筛期间被感染的人数减少，因此感病者的转移率 α_i 增加。这里将两者简化为正比关系。

4 问题分析

4.1 问题一分析

我们的目标是使得单轮核酸检测的总时长（包含检测时间和等待时间）最短。考虑到第一轮一旦出阳，第二轮的复查很可能需要分散（上门）检测，因此单次检测的平均时长要更长些，我们设为第一次混检的 c 倍。

基于这一假设，如果我们得到了本轮检测的阳性检出率 β （即一个人有多大概率是阳性），那么我们可以通过数学期望的方法得到平均每人的检测次数函数，通过求极值来获得混检率 K 。因此问题转化为如何获得较为合理的 β 。

由于数据有延迟，一个时刻的检出阳性数 (tested positive) 和实际感染者数 (infected patients) 往往是不一致的。我们仿照 J-M-W 模型^[4] 的想法，在假设较短时间内核酸检测的速度不变的情况下，用 $s(t)$ 表示 t 时刻核酸检测呈阳性的人数占比， $p(t)$ 表示 t 时刻感病者（包括检测出来的和未被检测出来的）的人数占比， τ 表示从感病到被检测出来是阳性的平均时间间隔。

如果 τ 恒定不变，那么我们很容易知道：

$$s(t) = p(t - \tau) \quad (1)$$

但是考虑到每个人的潜伏期并不相同，所以我们引入 G 表示该时间间隔 τ 的概率分布，给出：

$$s(t) = \int p(t - \tau') G(\tau', \tau) d\tau' \quad (2)$$

从而将求解阳性检出率 $\beta = s(t)$ 的问题，转化为了求解感病者人数占比 $p(t)$ 的问题。而后者可以通过收集单日新增和总感染数 (total infected number)，用数据拟合得到传染率 k ，再用 k 建立方程得到。

4.2 问题二分析

我们选取本次疫情中的两所代表性城市——上海和广州进行分析。首先，将累计的感染人数数据代入 J-M-W 模型，通过拟合获得传染率 k 和从感病到被检测出来是阳性

的时间间隔 τ , 从而得到阳性检出率 β 。接着通过数学期望模型求解出最理想的混检人数 K 。

4.3 问题三分析

在我们可以对多轮疫情统筹规划时, 一方面, 我们可以随着疫情的发展, 动态调节混检率 K , 使得检测更加高效; 另一方面, 我们也要注意由于 K 的改变也会对疫情走势有重大影响, 因此局部最优解 (单次检测) 未必是全局最优解。

比如说, 单次检测中较高的 K 可能效率较高, 但一管如果包含太多的人, 一旦出阳, 势必会使得第二次复查的速度减慢。换言之, 会使得病人在未封控的状态停留太久, 从而增加传染的风险。

因此, 本题的目标可以转换成在一轮疫情结束后, 使得多轮检测的总检测次数最小。解题的关键在于建立起可动态调节参数的传染病模型, 在每一轮检测过后, 调节参数; 在利用传染病模型的结果, 获得下一轮检测的最佳混检率; 由此形成循环前进, 直至疫情结束。

要想实现这一转染病模型, 传统时滞模型可作为重要参考, 但是对解决混检率帮助不大; 因此我们需要建立起二者的有机链接。

为了避免“贪婪算法”的问题, 还需要求模型有一定的“记忆功能”, 在多步选择中找到全局最优解。

5 模型建立与求解

5.1 问题一模型的建立

本题拟建立两个模型, 首先建立求得常微分方程的模型求得较为合理的阳性检出率 β , 随后利用求得的 β 建立数学期望模型求出最理想的混检人数 K 。

5.1.1 基于 J-M-W 时滞模型确定阳性检出率

用 $s(t)$ 表示 t 时刻核酸检测呈阳性的人数占比, $p(t)$ 表示 t 时刻感病者 (包括检测出来的和未被检测出来的) 的人数占比, τ 表示从感病到被检测出来是阳性的平均时间间隔, 那么我们有

$$s(t) = \int p(t - \tau') G(\tau', \tau) d\tau' \quad (3)$$

其中 G 是从感病到被检测出来是阳性的平均时间间隔 τ 的概率分布。一般地, 可假设 $G(\tau', t)$ 是一个正态分布:

$$G(\tau', t) = c_1 e^{-c_2(\tau' - t)^2}$$

于是给定某一时刻 t_0 , 我们就可以得到时刻 t_0 对应的阳性检出率

$$\beta = s(t_0) = \int p(t_0 - \tau') G(\tau', \tau) d\tau' \quad (4)$$

考虑到新冠病毒有潜伏期的存在，故从感病到被检测出来是阳性的时间间隔 τ 是不能被忽略的，所以这样求得的 β 是较为合理的。

接下来，只需考虑求解 $p(t)$ 即可。在时间 dt 内， $p(t)$ 的增加量 $dp(t)$ 为

$$dp(t) = kf(t)dt \quad (5)$$

其中， $f(t)$ 表示 t 时刻未被隔离起来的感病者的人数占比， k 是传染率，即单位时间内一个未被隔离起来的感病者能传染给多少个易感者，如果我们假设 t 时刻被隔离起来的感病者的人数占比为 $q(t)$ （包括核酸检测呈阳性的感病者和密切接触者），那么我们有等式

$$f(t) = p(t) - q(t) \quad (6)$$

我们进一步假设 $q(t)$ 与 $s(t)$ 是呈正比关系的，比例系数为 u ，即

$$q(t) = us(t) = up(t - \tau) \quad (7)$$

于是，我们得到了如下时滞微分方程：

$$\frac{dp(t)}{dt} = kf(t) = k[p(t) - q(t)] = k[p(t) - up(t - \tau)] \quad (8)$$

这是我们最终要求解的方程。该方程中有三个未知的系数：传染率 k ，平均隔离率 u 和从感病到被检测出来是阳性的时间间隔 τ 。其中前两者均可由收集单日新增感染者数量和感染者总数，用最小二乘法得到：

$$\min_{k,u} \|s(k,u) - s_{data}\|_2 \quad (9)$$

最后，我们通过给定 $p(0)$ 和某一时刻 t_0 就能得到较为合理的阳性检出率

$$\beta = s(t_0) = \int p(t_0 - \tau')G(\tau', \tau)d\tau' \quad (10)$$

5.1.2 混检期望模型的建立

在求得较为合理的阳性检出率 β 后，建立数学期望模型求得最佳的混检人数 K 。

假设采用 K 人混检的方式，用随机变量 X 表示每组中每个人需要的检测次数。由于每个样本可能被检测为阳性的概率为 β ，所以检测为阴性的概率是 $\gamma = 1 - \beta$ 。因为我们假设复查所耗费的平均时长更长，为第一次混检的平均时长的 c 倍，所以我们完全可以认为复查一次相当于 c 次。在以上假设之下，随机变量 X 的取值情况如下：

$$X = \begin{cases} \frac{1}{K} & \text{该组混合样本结果是阴性} \\ \frac{1}{K} + c & \text{该组混合样本结果是阳性} \end{cases}$$

随机变量 X 的分布规律如下表所示：

X (取值)	$\frac{1}{K}$	$\frac{1}{K} + c$
P (概率)	γ^K	$1 - \gamma^K$

表 2: X 的分布规律

故随机变量 X 的数学期望

$$E(X) = \frac{1}{K}\gamma^K + \left(\frac{1}{K} + c\right)(1 - \gamma^K) = c + \frac{1}{K} - c\gamma^K \quad (11)$$

为了使核酸检测效率最大化，应该使每组中每个人需要检测次数的数学期望最小。于是，问题转化为：在 γ 给定的情况下，求正整数 K 的值，使得 $E(X)$ 最小。

5.2 问题二数据的来源与加工

我们选取本次疫情中的两所代表性城市——上海和广州进行分析。首先，将累计的感染人数数据代入 J-M-W 模型，数据来源于上海卫生健康委员会和广州卫生健康委员会^[1,2]。从数据中，我们可以获得每天的新增感染人数 $N\Delta p(t) = N(p(t+1) - p(t))$ ，进而可以求得累计感染人数 $Np(t) = N \sum_{i=0}^{t=t} \Delta p(i)$ 。接着，利用 J-M-W 模型中的方程：

$$\frac{dp(t)}{dt} = k[p(t) - up(t-\tau)] \quad (12)$$

我们便可通过求解最小二乘法，获得传染率 k ，以及初始时刻的阳性检出率 $\beta_1 = p(t_0-\tau)$ 。此外，按照定义，治愈率 h 的计算式为：

$$h = \frac{l}{H} \quad (13)$$

其中 l 为一天内出院的人数， H 为累计在院治疗的人数。 l 和 H 的数据均来源于上海卫生健康委员会和广州卫生健康委员会。治愈率 h 取近 1 至 2 周内的平均值。

5.3 问题三模型的建立

这是一个涉及到疫情发展的动态问题。为了和已有的传染病模型建立起联系，基于假设 (H2)，我们可以把目标从“使得总核酸检测次数最少”转化为“尽可能快地完全控制住疫情”（事实上二者是一样的），即最小化 r 轮核酸检测的总时间 $T = t_1 + t_2 + \dots + t_r$ 。

5.3.1 局部最优模型

传统的时滞动力学模型，需要解带有固定时滞的时滞微分方程 (DDE)，不仅较难求解 (matlab 的 dde23 程序运行较慢)，且也对本题求解混检率没有助益。考虑到时滞模型最关键的是实现阳性检出率与感染率的转换（因为新冠病毒善于潜伏），我们试图探索一种模式，既免去解 DDE 方程组，又能获得混检率 K 。

具体来说，我们首先通过每次测核酸的间隔，将整个疫情时间轴分为一个个周期。在每个周期伊始，先用阳性检出率计算出最佳混检率 K ，并利用时滞假设，计算得到感染率 $k_1 k_2$ ；然后，把感染率代入到一般的 SIR（本文为 SIJ）中得到各类人群的变化情况；最后，利用周期末端的人群分布情况，利用时滞假设更新阳性检出率。

因此，首先我们构建基于 SIR 模型的 SIJ 模型，其中 S 代表易感者，感病者一旦被确诊，通常会被隔离，其传染性显然比未被确诊的感病者要小得多，因此可以将感病者分为两类：未确诊的感病者（仍称为感病者，记为 I）和已确诊的感病者（称为确诊者，记为 J）由于新冠患者在痊愈后仍会再次患上新冠，以及目前死亡率较低，故不考虑存在移出者 R（即那些患病后不会再次患病而的人和死者）。我们有：

$$\begin{cases} \frac{dS}{dt} = -(k_1 I + k_2 J)S + h_1 I + h_2 J \\ \frac{dI}{dt} = (k_1 I + k_2 J)S - (\alpha + h_1)I \\ I + J + S = 1 \\ S(0) = S_1, I(0) = I_1, J(0) = J_1 \end{cases} \quad (14)$$

其中 k_1, k_2, h_1, h_2 分别是 I 类病人和 J 类病人的传染率和治愈率， α 是转移率。

下面我们来构建我们的“检测块”模型。称两次检测之间的间隔为一个周期。

在第一个周期，采用第一问的方法拿到起始数据 $S_1, I_1, J_1, \alpha_1, k_1(1), k_2$ 。利用第一题的模型，拿到最优混检率 K_1 。设城市人口为 N ，每日核酸检测速率为 v ，则可以得到第一次核酸检测的总时长为

$$t_1 = \frac{N(c + 1/K - c\gamma^K)}{v} \quad (15)$$

将 $[0, t_1]$ 作为积分区间求解我们的 SIJ 模型，得到该阶段最后的 S,I,J 分布比例。

在第 i 个周期 ($i \geq 2$) 内，我们需要基于上一周期的结果更新参数，其中积分末端点的函数值为下一个周期的初始值，即

$$S_i = S_{last}(t_{i-1}), I_i = I_{last}(t_{i-1}), \beta_i = \int (1 - S_i(t - \tau')) G(\tau', \tau) d\tau' \quad (16)$$

其中 S_{last} 和 I_{last} 为第 $i-1$ 周期内解得的积分函数。考虑到混检率 K 对于疫情走势的影响，转移率 α 采用如下方法更新：

$$\alpha_i = \frac{\gamma^{K_{i-1}} N}{v t_{i-1}} \alpha_1 \quad (17)$$

系数的物理含义是复筛阶段占周期的比例，因为我们认为，如果复筛所耗的时间比较长，那么阳性的病人在等待过程中并未得到转移，这会导致转移率的下降。

由于已封控病人的传染率 k_2 本来就不高，所以我们认为其不随轮数 i 的变化而变化，即认为它是一个常数。基于未封控病人的传染率 k_1 在短时间（相邻天）不变的假设，我们可考虑用积分区间末端点前一天的传染率来作为下一个周期的 k_1 。从差分法的思想出发，就得到如下等式：

$$k_1(i+1) = (S_{last}(t_{i-1}) - S_{last}(t_{i-1}-1)) / S_{last}(t_{i-1}-1) \quad (18)$$

记录每个周期的 K_i 和时间长 t_i 。判断 S_i 是否大于 0.99999（疫情是否结束），若未结束，则进入下个周期，重复这个过程，流程图如下所示：

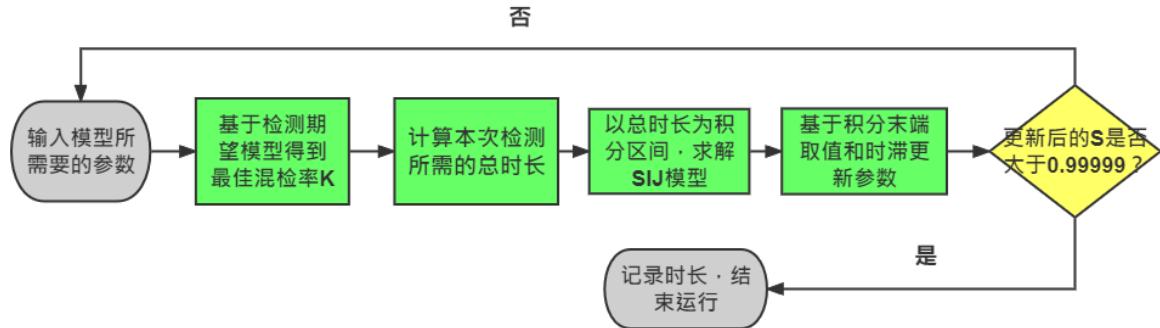


图 1: 局部最优流程图

5.3.2 从局部最优逼近全局最优

上述方法是用贪婪算法得到每个周期内的局部最优解，但他未必是全局最优解。为了解决这一问题，我们引入束搜索的办法。

具体来说，在第 i 个周期，我们得到局部最佳的 K_i 后，考虑到目标函数的连续可微性， K_{i-1} 和 K_{i+1} 也是相对比较好的答案。我们把他们都储存下来，分别计算接下来三个周期的时间和（若疫情结束，则后续周期时长取 0），把总和最短的那个记为真正的 K_i 。

这种做法既适当缓解了贪婪算法的坏处，又避免了穷举算法计算量过大，程序卡死的问题。我们把保留的可选解个数记为束宽，探索的周期数记为束长，下面这副示意图展示了束宽为 2，束长为 3 的情况。

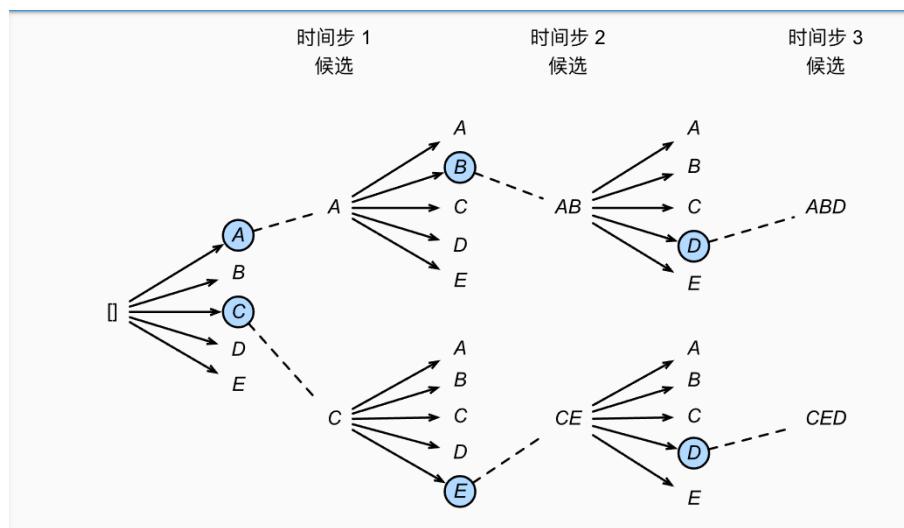


图 2: 束搜索示意图

6 模型的求解与结果分析

6.1 参数的求解

本次疫情的时间自 2022 年 3 月开始。自三月中旬开始，由于上海的感染人数的急增，上海政府决定进行封城。而广州的疫情相对较轻，因此尚未采取封城的措施。我们将 4 月份的疫情数据采用 Levenberg-Marguad (L-M) 方法求解最小二乘法问题，反演得到最优解 k ，将由 k 得到的曲线与 4 月上海和广州的疫情曲线作对比，发现和实际数据拟合较好。

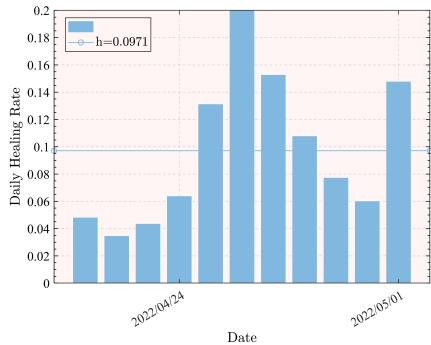


图 3: 治愈率 (上海)

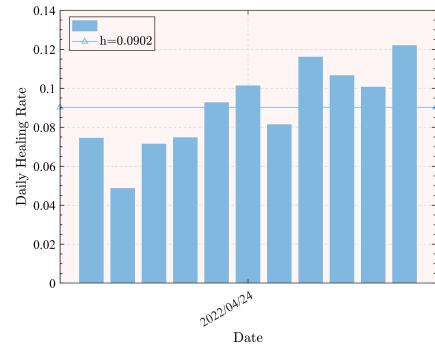


图 4: 治愈率 (广州)

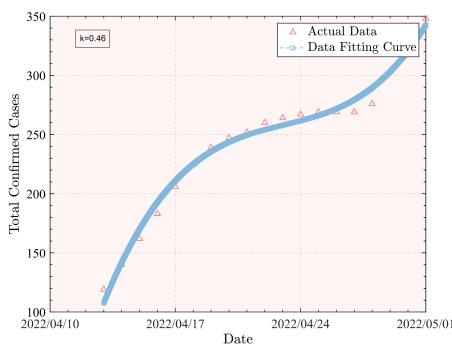


图 5: 病例总数 (上海)

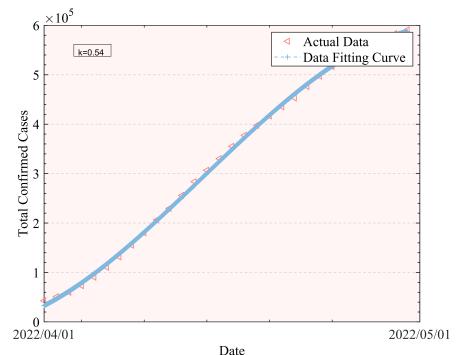


图 6: 病例总数 (广州)

6.2 问题一模型的求解

沿问题一中的记号，我们发现，对于上海，拟合得到时间间隔 $\tau_{Shanghai} = 1$ (天)， $k_{Shanghai} = 0.54$ ；对于广州，拟合得到时间间隔 $\tau_{Guangzhou} = 2$ (天)， $k_{Guangzhou} = 0.49$ 。再利用式 (8)，得到上海和广州的阳性检出率分别为 $\beta_{Shanghai} = 0.983\%$ 和 $\beta_{Guangzhou} = 0.157\%$ 。

由于检测为阴性的概率是 $\gamma = 1 - \beta$ ，因此 $\gamma_{Shanghai} = 99.017\%$, $\gamma_{Guangzhou} = 99.843\%$ 。根据相关的新闻报道和现实情况，我们取比例系数 $c = 1.5$ ，我们计算得每个人需要检测次数 $E(x)$ 与混检人数 K 的关系如下表所示。

K	$E(x)$
6	0.25299
7	0.24310
8	0.23898
9	0.23872
10	0.24110
11	0.24536
12	0.25101

表 3: 上海的情况

K	$E(x)$
18	0.09738
19	0.09675
20	0.09640
21	0.09631
22	0.09642
23	0.09672
24	0.09718

表 4: 广州的情况

可以看出,为了使检测次数 $E(x)$ 最低,混检人数应当取为 $K_{Shanghai} = 9, K_{Guangzhou} = 21$ 。在这种情况下, 总体核酸检测速度是最快的, 从而最大程度地减少了因核酸检测而耽误的控制病毒传播的宝贵时间。

6.3 问题三模型的求解

在 SIJ 模型中, 我们需要区分 I 类病人和 J 类病人不同的传染率。为此, 依据白云, 2022 等人的工作, 我们可以假设广州 I 类病人的传染率是 J 类病人的 5 倍。其中考虑到上海 4 月 1 号后已封城, 未封控群体实际居家隔离, 因此假设 I 类病人的传染率是 J 类病人的 2 倍。把问题一拟合得到的 k 加权分配, 就得到 k_1 和 k_2 。

把参数代入数据, 我们分别模拟上海和广州基础转移率较高 ($\alpha_1 = 0.3$) 和较低 ($\alpha_1 = 0.02$) 的情况, 所得疫情的发展情况如下图所示:

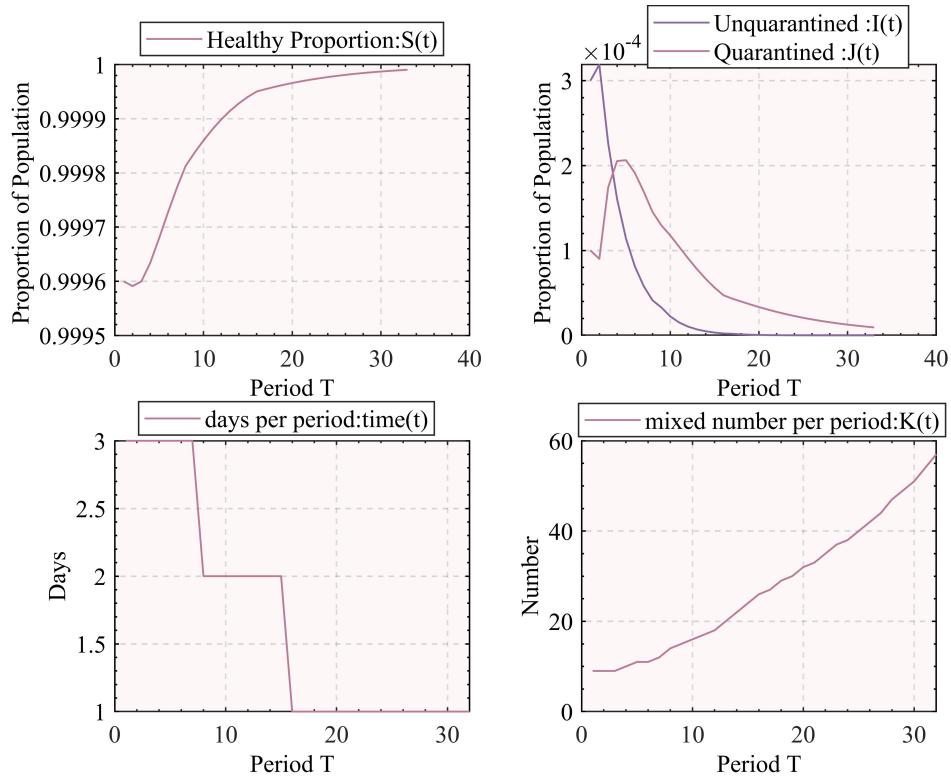


图 7: 上海 ($\alpha_1 = 0.02$)

注: 左上为易感人群 S 随周期变化图, 右上为 I, J 变化图; 左下为每周期的时长 t ; 左下为每周期的混检率 K , 广州图含义与上海一致。

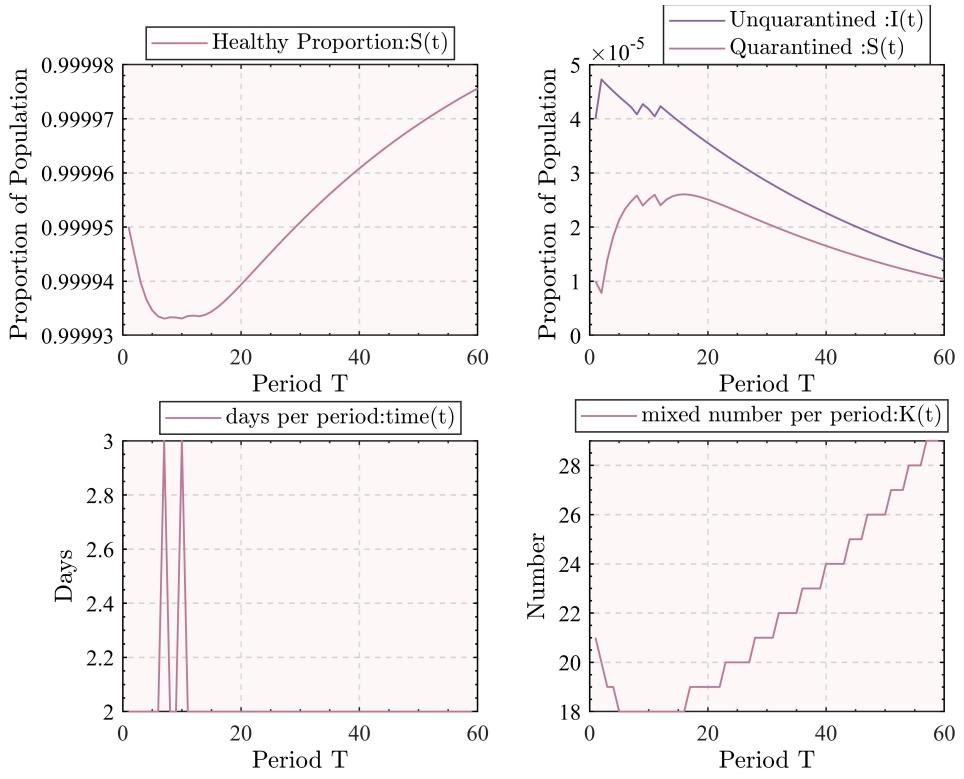


图 8: 广州 ($\alpha_1 = 0.02$)

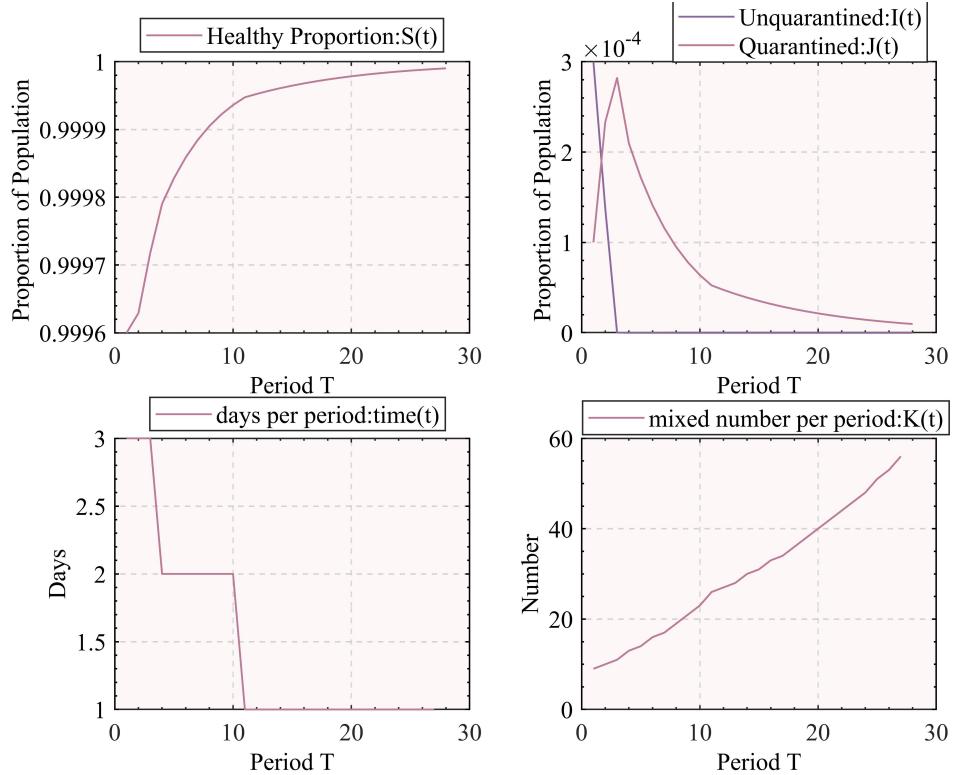


图 9: 上海 ($\alpha_1 = 0.3$)

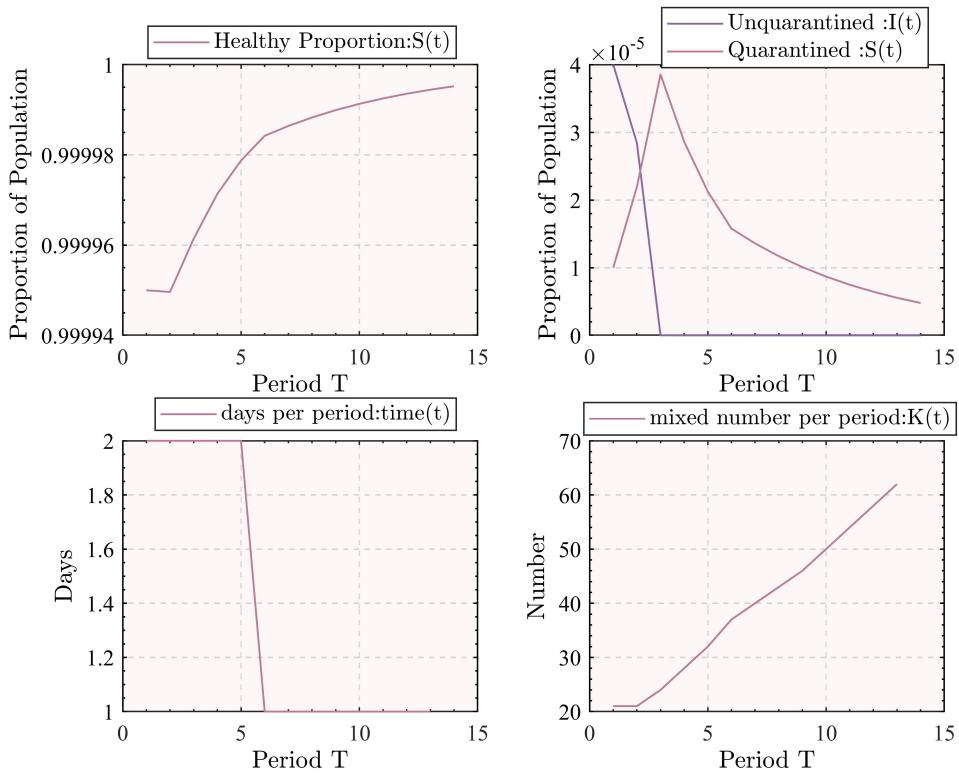


图 10: 广州 ($\alpha_1 = 0.3$)

混检率变化如下表所示:

表 5: 上海混检率一览图

转移率/周期	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
转移率 =0.02	9	9	9	10	11	12	12	14	16	17	17	18	19	20	20	22
转移率 =0.3	9	10	11	13	14	15	16	18	20	22	25	26	28	29	31	32
转移率/周期	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
转移率 =0.02	23	23	24	25	27	29	33	36	38	40	42	47	49	49	51	53
转移率 =0.3	34	36	38	40	42	44	46	49	52	55						

表 6: 广州混检率一览表 (转移率为 0.02 时疫情至 32 周期仍未结束)

转移率/周期	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
转移率 =0.02	21	20	19	19	19	19	19	19	20	20	20	20	21	21	22	22
转移率 =0.3	21	21	24	28	32	37	40	43	46	50	54	58	62			
转移率/周期	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
转移率 =0.02	22	23	23	24	25	25	25	26	27	27	28	28	29	30	31	33
转移率 =0.3																

从结果图表中我们可以发现:

1. 混检率会随着疫情的严重程度而变化，疫情越严重，混检率越低，这和我们的直观影响一致。

2. 转移率 α 对疫情走势影响很大。从上海的情况而言，若转移率较高（基础转移率为 0.3），那么不仅健康人群比例逐步提高，且抗疫总周期越短（意味着每人平均测核酸次数越少），共 $3 \times 2 + 2 \times 7 + 1 \times 16 = 36$ 天；相反，若转移率较低（基础转移率为 0.02），不仅防疫前期疫情会更加严重，（共有 $3 \times 3 = 9$ 天感染人数增多，且防疫总周期高达 50 天之久。由于广州未封控人群与已封控人群的感染率之比更大，如果转移率较低，甚至会导致疫情在 60 个周期内仍无法结束的窘境。

3. 由束搜索导致的最优解改变在疫情高峰时段影响较大。下表是上海是否采取束搜索的混检查率之比。

表 7: 束搜索与贪婪算法结果对比图 (上海, $\alpha_1=0.02$)

方式	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
束搜索	9	9	9	10	11	12	12	14	16	17	17	18	19	20	20	22
局部最优	9	10	10	11	12	13	13	15	16	17	17	19	19	20	20	22
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
束搜索	23	23	24	25	27	29	33	36	38	40	42	47	49	49	51	53
局部最优	23	23	24	25	27	29	33	36	38	40	42	47	49	49	51	53

从表格中我们可以看出，只有疫情早期，鉴于未封控感染人数较多，从提高转移率的角度出发，束搜索会抑制混检率的增长。

4. 每次检测周期会随着疫情逐渐向好而缩短。

5. 在疫情后期，最佳混检率较高，从技术而言似乎无法达到 50: 1 的混采模式。其实，模型做出这样的结果主要是因为当疫情即将被扑灭时，并没有必要再进行大范围的全城核酸检测。基于数据，可建议政府在疫情后期仅在重点区域进行核酸检测，否则过度耗费医疗资源，得不偿失。

6.4 灵敏度分析

尽管我们利用 J-M-W 模型的假设，通过式子 (2) 将阳性检出率与过去某个时刻的感染率建立了联系，但是鉴于新冠疫情的传播手段多样，且大城市情况复杂，所获得的初始阳性检出率理论值 β_1 可能和实际有一定的误差。为此，我们测试在值 β_1 上下浮动时，模型结果是否能保持稳定。

如下图所示，我们把 1.2 倍理论值 β_1 所得结果记为 More 类，把 0.8 倍理论值 β_1 所得结果记为 Less 类。可以看到，无论是增大还是减小 β_1 ，均不会对结果产生太大的影响；因此，初始参数的扰动不会影响模型的精确性。

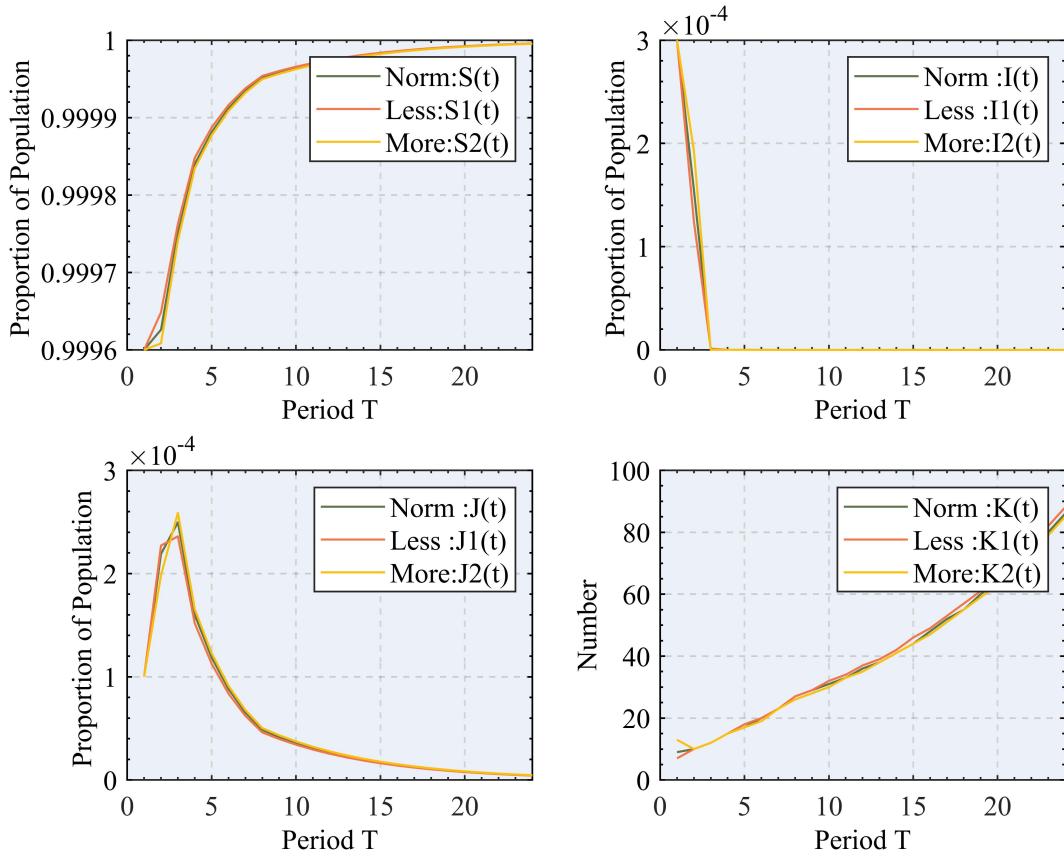


图 11: 上海 (转移率为 0.3) 在 β_1 浮动时的变化情况

7 模型评价与改进

本文针对核酸检测的混检方案问题进行了数据分析和数学建模。我们收集了上海市和广州市 2022 年 4 月份的疫情感染人数数据，基于该数据，通过数据拟合以及数学建模的方法，给出了单轮和多轮核酸检测的混检方案。我们的模型优势主要体现在：

- (1) 基于新冠病毒潜伏期长的特点，引入时滞模型。一般而言，我们能获得的数据来源只有某时刻的阳性检出人数，这显然和实时感染人数有一定的差别；我们基于时滞思想，建立了二者的数学关系，并且发现和数据对比拟合的很好。
- (2) 基于目标是混检率，改进传统的时滞动力学系统。我们引入检测周期的概念，在周期内使用传统的 SIJ 模型，而周期间则利用时滞属性更新参数。
- (3) 核心思想简单。主要利用概率学的模型，刻画了核酸检测的效率与混检人数之间的关系。本文避免了使用蒙特卡洛模拟方法，以明确、清晰的数学方法计算了核酸检测所花费的总时间与混检人数的关系，结论深刻。
- (4) 灵活性高。对于不同的城市，只需要对代码中的数据代码进行部分修改就可以进行计算。代码运行速度较快，且寻找到的最优解具有现实的参考意义。

我们的模型不足主要为：

- (1) 模型使用的近似次数较多，求出的精准度可能有一定欠缺。
 - (2) 模型在阳性检出率较低的情况下得到的结果可能会得到过高的混合数，不符合实际显示情况。这一点可通过分区使用不同的混检率，或者取消全域核酸检测得到缓解。
- 我们模型未来可以从以下几点进行改进：
- (1) 对现有的模型做出修正，减少近似的次数，提高精准度。
 - (2) 纳入更多现实因素进行考虑，以实现更强的现实性。

8 给政府的建议

1. 当中国进行全城范围内的核酸大筛查时，通常采取 10 人以上的混检都是可取的。这是因为，由于中央始终坚守动态清零策略，故即便在上海，健康人群的比例都要超过 99%；根据我们的模型，最佳混检率都在 10 人以上。当然，由于国外疫情较为严重，恐怕不能采用这种节约成本的方式。
2. 必须要有效提高未封控病人到封控病人的转移率。无论是上海还是广州，如果不能够及时地把病人筛选出来并且统一封控，那么都会造成疫情清零日期延后，甚至会出现无法控制的窘境。这一点在未实施静态管理的区域（如广州）尤为关键。
3. 在疫情后期，考虑到病人已经相对较少，阳性检出率在全域范围内较低。这时，可考虑针对风险等级不同的区域，采取不同的混检率，以提高资源利用效率。在风险等级较低的区域，也可以降低核酸检测频率，既不打扰百姓，也不会因为多人混采技术无法做到 50：1 混采而达不到理论上的最优解。

参考文献

- [1] 上海市卫生健康委员会, 新闻发布,
<https://wsjkw.sh.gov.cn/xwfb/index.html>,2022 年 5 月 3 日.
- [2] 广州市卫生健康委员会, 疫情通报,
<http://wjw.gz.gov.cn/ztzl/xxfyqfk/yqtb/index.html>,2022 年 5 月 3 日.
- [3] 陈本晶, 陈敏娜, 基于新型冠状病毒核酸混合检测的数学模型及其意义, 新一代信息技术, 4(23):31-35,2021.
- [4] Minghao Jiang,Guoyu Meng,Geng Wu,
The Dawn is Coming ——the Description and Prediction of Omicron SARSCoV-2 Epidemic Outbreak in Shanghai by Mathematical Modeling,
<https://www.medrxiv.org/content/10.1101/2022.04.13.22273788v2>,2022 年 5 月 2 日.

- [5] 白云, 钱箴, 孙玉莹, 汪寿阳, 基于综合集成预测方法的新冠肺炎疫情预测, 系统工程理论与实践:1-19,2022.
- [6] 董章功, 宋波, 孟友新, 基于 SEIR-ARIMA 混合模型的新冠肺炎预测, 计算机与现代化 (02):1-6,2022.
- [7] 严阅, 陈瑜, 刘可^F, 罗心悦, 许伯熹, 江渝, 程晋, 基于一类时滞动力学系统对新型冠状病毒肺炎疫情的建模和预测, 中国科学: 数学,50(03):385-392,2020.

9 附录

模型主代码 (matlab) :

```
clc
clear
close all;
h1=0.001;v=2000000;N=25000000;
h2=0.15;
z1=0.001;z2=0.001;
p=zeros(60,1);I=zeros(60,1);S=zeros(60,1);a=zeros(60,1);t=zeros(60,1);
k=zeros(60,1);J=zeros(60,1);
kspr1 = zeros(60, 1); kspr2 = zeros(60, 1);
kspr1(1) = 0.08; kspr2(1) = 0.001;
a(1)=0.3;
I(1) = 0.0003;
S(1) = 0.9996;p(1)=(1-S(1))*20;
J(1)=1-I(1)-S(1);
i=0;if = 0;
while (S(i+1)<0.999996) (i<59)
i=i+1;
plocal = p(i); Slocal = S(i); Ilocal = I(i); alocal = a(i);
ksprloc1 = kspr1(i); ksprloc2 = kspr2(i);
fun = @(x)1.5+1/x-1.5*(1-plocal)x;
x0 = 3;
[klocal1, eval] = fminunc(fun, x0);
klocal = round(klocal1); k(i) = klocal;
M=N*(1.5+1/klocal - 1.5 * (1 - plocal)^(klocal)); t(i) = ceil(M/v);
a(i+1)=2*(1-plocal)ikocal * N/(v * t(i)) * a(1);
tloc = t(i);
tspan = [0 tloc];
y0 = [SlocalIlocal];
[t1, y] = ode45(@(t,y)odefun(t,y,ksprloc1, ksprloc2, h1, h2, alocal), tspan, y0);
Step=max(0,y(end,1));S(i+1)=min(1,Step);
Itep=max(0,y(end,2));I(i+1)=min(1-Step,Itep);
J(i+1) = max(0,1-S(i+1)-I(i+1));
p(i+1)=min(0.012-(i/6000),(1-S(i+1))*20);
kspr1(i + 1) = kspr1(i); kspr2(i + 1) = kspr2(i);
```

```

end
T=zeros(60,1);
for j=1:60
T(j)=j; end
I_f = zeros(60, 3); S_f = zeros(60, 3); t_f = zeros(60, 3); k_f = zeros(60, 3); J_f =
zeros(60, 3);
for tot=1:i
I_f(tot, 1) = I(tot); S_f(tot, 1) = S(tot); J_f(tot, 1) = J(tot);
t_f(tot, 1) = t(tot); k_f(tot, 1) = k(tot);
end
i_f = i;
i=0;p(1)=(1-S(1))*20*2; while (S(i+1)<0.999996) (i<59) i=i+1; p_local = p(i); S_local =
S(i); I_local = I(i); a_local = a(i); k_sprloc1 = k_spr1(i); k_sprloc2 = k_spr2(i); fun = @(x)1.5+
1/x-1.5*(1-p_local)x; x0 = 3; [k_local1, eval] = fminunc(fun, x0); k_local = round(k_local1); k(i) =
k_local; M = N*(1.5+1/k_local - 1.5*(1-p_local)(k_local)); t(i) = ceil(M/v); a(i+1) = 2 *
(1-p_local)k_local*N/(v*t(i))*a(1); t_loc = t(i); tspan = [0t_loc]; y0 = [S_local I_local]; [t1, y] =
ode45(@(t, y)odefun(t, y, k_sprloc1, k_sprloc2, h1, h2, a_local), tspan, y0); Step = max(0, y(end, 1)); S(i+1) =
min(1, Step); Itep = max(0, y(end, 2)); I(i+1) = min(1 - Step, Itep); J(i+1) =
max(0, 1 - S(i+1) - I(i+1)); p(i+1) = min(0.012 - (i/6000), (1 - S(i+1))*20); k_spr1(i+1) =
k_spr1(i); k_spr2(i+1) = k_spr2(i); endforj = 1 : 60T(j) = j; endfor tot = 1 : i
I_f(tot, 2) = I(tot); S_f(tot, 2) = S(tot); J_f(tot, 2) = J(tot); t_f(tot, 2) = t(tot); k_f(tot, 2) =
k(tot); endif i_f > i; if i_f = i; endi = 0; p(1) = (1 - S(1)) * 20 * 0.5; while(S(i + 1) <
0.999996)(i < 59)i = i+1; p_local = p(i); S_local = S(i); I_local = I(i); a_local = a(i); k_sprloc1 =
k_spr1(i); k_sprloc2 = k_spr2(i); fun = @(x)1.5+1/x-1.5*(1-p_local)x; x0 = 3; [k_local1, eval] =
fminunc(fun, x0); k_local = round(k_local1); k(i) = k_local; M = N*(1.5+1/k_local - 1.5 *
(1-p_local)(k_local)); t(i) = ceil(M/v); a(i+1) = 2 * (1-p_local)k_local*N/(v*t(i)) *
a(1); t_loc = t(i); tspan = [0t_loc]; y0 = [S_local I_local]; [t1, y] = ode45(@(t, y)odefun(t, y, k_sprloc1, k_sprloc2),
max(0, y(end, 1))); S(i+1) = min(1, Step); Itep = max(0, y(end, 2)); I(i+1) = min(1 -
Step, Itep); J(i+1) = max(0, 1 - S(i+1) - I(i+1)); p(i+1) = min(0.012 - (i/6000), (1 -
S(i+1))*20); k_spr1(i+1) = k_spr1(i); k_spr2(i+1) = k_spr2(i); endforj = 1 : 60T(j) =
j; endfor tot = 1 : i
I_f(tot, 3) = I(tot); S_f(tot, 3) = S(tot); J_f(tot, 3) = J(tot); t_f(tot, 3) =
t(tot); k_f(tot, 3) = k(tot); endif i_f > i; if i
subplot(2,2,1);
plot(T(1:i_f), S_f(1 : i_f, 1), '-o', T(1 : i_f), S_f(1 : i_f, 2), '-o', T(1 : i_f), S_f(1 : i_f, 3), '-o');
legend('Norm:S(t)', 'Less:S1(t)', 'More:S2(t)');
ylabel('Proportion of Population ');
xlabel('Period T');

```

```

hold on;
subplot(2,2,2)
plot(T(1:i),I_f(1 : i, 1),T(1 : i),I_f(1 : i, 2),'g',T(1 : i),I_f(1 : i, 3));
legend('Norm :I(t)','Less :I1(t)', 'More:I2(t)', 'Best');
ylabel('Proportion of Population ');
xlabel('Period T');

hold on;
subplot(2,2,3)
plot(T(1:i),J_f(1 : i, 1),T(1 : i),J_f(1 : i, 2),'g',T(1 : i),J_f(1 : i, 3));
legend('Norm :J(t)', 'Less :J1(t)', 'More:J2(t)', 'Best');
ylabel('Proportion of Population ');
xlabel('Period T');

hold on;
hold on; subplot(2,2,4)
plot(T(1:i),k_f(1 : i, 1),'-.',T(1 : i),k_f(1 : i, 2),'-.',T(1 : i),k_f(1 : i, 3),'-.');
legend('Norm :K(t)', 'Less :K1(t)', 'More:K2(t)', 'Best');
ylabel('Number');
xlabel('Period T');

function dydt = odefun(t,y,k1,k2,h1,h2,a)
dydt = zeros(2,1);
dydt(1) = -(k1*y(2)+(1-y(1)-y(2))*k2)*y(1)+h1*y(2)+h2*(1-y(1)-y(2));
dydt(2) = (k1*y(2)+k2*(1-y(1)-y(2)))*y(1)-(a+h1)*y(2);
end

```