HOCHSCHULE
MITTWEIDA
University of Applied Sciences

BACHELOR THESIS

Mr.
Dmitrii Rudanov, B.Sc.

Transmission Dynamics of a COVID-19 Outbreak in a Homeless Shelter in Chicago, Illinois, USA

Mittweida, November 2023

Faculty of **Applied Computer Sciences and Biosciences**

BACHELOR THESIS

Transmission Dynamics of a COVID-19 Outbreak in a Homeless Shelter in Chicago, Illinois, USA

Author:
Dmitrii Rudanov

Course of Study:
Applied Mathematics

Seminar Group:
MA20w1-B

First Examiner:
Prof. Dr. rer. nat. habil. Kristan Schneider

Second Examiner:
Prof. Dr. rer. nat. Franka Baaske

Submission:
Mittweida, 07.11.2023

Defense/Evaluation:
Mittweida, 2023

Bibliographic Description

Rudanov, Dmitrii:
Transmission Dynamics of a COVID-19 Outbreak in a Homeless Shelter in Chicago, Illinois, USA. – 2023. – 23 S.
Mittweida, Hochschule Mittweida – University of Applied Sciences, Faculty of Applied Computer Sciences and Biosciences, Bachelor Thesis, 2023.

Referat

In this thesis, we implement, correct, and modify the compartmental model described in "Transmission Dynamics of Large Coronavirus Disease Outbreak in Homeless Shelter, Chicago, Illinois, USA, 2020" [5]. Our objective is to engage in reading and understanding scientific literature, reproduce the results, and modify or generalize an existing mathematical model. We provide an overview of epidemiological models, focusing on simple compartmental SEIR models. We correct inaccuracies and misprints in the original implementation and use the limited-memory Broyden–Fletcher–Goldfarb–Shanno [13] algorithm to fit the model's parameters. Furthermore, we modify the model of [5] by introducing an additional compartment. The resulting model has a more intuitive interpretation and relies on fewer assumptions. We also perform the fitting process for this alternative model. Finally, we demonstrate the advantages of our modified implementations and discuss other possible approaches.

Contents

List of Figures

List of Tables

Source code

Acknowledgment

We would like to express our deepest appreciation to Prof. Schneider, whose invaluable guidance has shaped this research. His profound expertise and insightful feedback have been instrumental in the completion of this paper. His dedication and commitment to fostering a rich learning environment have been truly inspiring. We are grateful for his time, patience, and unwavering support throughout this journey.

1 Introduction

The aim of this thesis is to implement and modify the model of the coronavirus disease 2019 (COVID-19) outbreak in a homeless shelter in Chicago, Illinois, USA, in 2020, which was described in [5]. The objective is to read and understand scientific literature, reproduce the results, and modify or generalize an existing mathematical model.

The emergence of the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) causing COVID-19 has led to a pandemic with a profound impact on the world, affecting billions of people and causing widespread disruption to daily life. The virus was first identified in Wuhan, China in December 2019 and has since spread to become a global pandemic. This led the WHO to declare a *Public Health Emergency of International Concern* (PHEIC) on 30 January 2020, and to characterize the outbreak as a pandemic on 11 March 2020 [17].

For epidemiological prediction and to support decision-making, mathematical models are standard tools in epidemiology. Such models are based on a variety of factors, including the biology of the disease, the characteristics of the population, and the effectiveness of interventions such as social distancing and vaccination. Predictions are utilized to guide decisions in public health regarding planning for pandemics, allocating resources, implementing measures for social distancing, and other interventions [6].

1.1 Biology of the virus

COVID-19 is caused by SARS-CoV-2, which belongs to the broad family of viruses known as coronaviruses. These are positive-sense, single-stranded RNA viruses (Baltimore Group IV). SARS-CoV-2 is an enveloped β-coronavirus, with a genetic sequence very similar to SARS-CoV-1 (80%) and bat coronavirus RaTG13 (96.2%). The relatively large genome (29.7knt) codes, like all coronaviruses, for four structural proteins, the nucleocapsid protein (N), spike (S) glycoprotein coating the viral envelope, and the envelope (E) and membrane (M) proteins [3].

The virus primarily invades the respiratory system and gains entry into human cells via the angiotensin-converting enzyme 2 (ACE2) receptor. Upon gaining entry, the viral RNA is inserted into a host cell, which uses the host cell's replication mechanism to multiply. The host cell releases new virions that infect other cells [3].

The peak SARS-CoV-2 load in the respiratory tract is observed at the time of symptom onset or in the first week of illness, with subsequent decline thereafter, indicating the highest infectiousness potential just before or within the first five days of symptom onset [3].

1.2 Disease spread and symptoms

SARS-CoV-2 is an airborne virus, which primarily is transmitted through the air are inhaled at short range (short-range aerosol or short-range airborne transmission). The virus can also spread by touching a surface or object contaminated with the virus and then touching

one's mouth, nose, or eyes. The virus is especially spreading well in enclosed spaces with poor ventilation. It is important to note that people who are infected with COVID-19 can be contagious even if they do not have symptoms [18].

The symptoms of COVID-19 can vary from person to person, but the most common symptoms include fever or chills, cough, shortness of breath or difficulty in breathing, fatigue, muscle or body aches, headache, loss of taste or smell, sore throat, congestion or runny nose, nausea, or vomiting. It is important to note that not everyone with COVID-19 will experience all of these symptoms, and some people may not have any symptoms at all [8]. However, COVID-19 can be severe, if the virus enters the lower respiratory tract. Individuals with an intense infection need hospitalization or even ICU treatment. Risk factors associated with severe COVID-19 cases are age, obesity, and co-morbidities like diabetes, HIV, etc. [7]. Until now, almost 7 million deaths worldwide have been attributed to COVID-19 [22].

1.3 Epidemiological models and evaluation

Many approaches to epidemiological models exist, such as deterministic (SIR-type), individual-based, and stochastic models. The model and the approach need to be tailored to a specific situation to be accurate [16].

Evaluating epidemiological models, such as for the spread of SARS-CoV-2 involves several steps and considerations:

- **Understanding the model type**: Models can be deterministic or stochastic. Deterministic models as those considered here are typically compartmental models based on the SIR (susceptible, infected, recovered) model. They typically vary in their complexity and include additional compartments such as exposed or asymptomatic individuals, or age groups [21]. Whether a determinist or stochastic model is more appropriate depends on the magnitude of the disease outbreak and the phase to be studied. In general, due to the stochastic nature of a disease outbreak, stochastic models are preferable at the onset of an epidemic, while deterministic models are more convenient to model an ongoing epidemic outbreak in a sufficiently large population in which random effects can be ignored [2].
- **Estimating key parameters**: Important parameters include the basic reproduction number R_0, which indicates how many people an infected person will infect on average, the case fatality rate, which is the proportion of infected people who die from the disease, the incubation period, the duration of various phases of the disease, the contact behavior, etc. [21]. Ideally, a model is set up in such a way that parameters can be estimated intuitively.
- **Assessing assumptions**: Models require assumptions about the biology of the disease (mode of transmission, transmissibility, etc.) which must be based on biological facts about the underlying pathogen. Model assumptions should be in accordance to empirical evidence of past and current disease outbreaks. Empirical evidence includes facts about the incubation period (i.e., the time between infection and disease outbreak) and latency periods (i.e., the time between infection and infectiousness), the contact

behavior, rates of transmission among different age groups, and the number of people who are immune [21]. These assumptions should be reasonable and based on available data.

- **Comparing Predictions with Real-World Data**: Models should be validated by comparing their predictions with actual case data. Discrepancies can help to refine the model [21].

It is important to remember that while models can provide important insights, they are simplifications of reality and their predictions are subject to uncertainty. Therefore, their results should be interpreted carefully.

1.4 Background

Persons living in congregate settings, including homeless shelters, have been disproportionately affected by the COVID-19 pandemic [10]. Individuals who are homeless are more susceptible to SARS-CoV-2 infection due to their communal living conditions and challenges in maintaining physical distancing. Additionally, they are more likely to experience severe COVID-19 symptoms because of the higher occurrence of their underlying health condition [9].

In this thesis, we implement and modify the model from [5]. Community transmission was documented in Chicago, Illinois, USA, in early March, and a statewide stay-at-home order was implemented on March 14, 2020. From March to May 2020, many homeless shelters in Chicago experienced COVID-19 outbreaks [10].

We reproduce the compartmental mathematical model of an outbreak of COVID-19 in Chicago's largest homeless shelter, and correct errors and inaccuracies in the original article [5]. Furthermore, we introduce an improved model that allows different incubation periods for symptomatic and asymptomatic individuals. The advantage of this extension is that the model interpretation is more intuitive and the extended models can be more easily parametrized.

2 Methods

Here, we introduce the background and model of [5]. The presentation follows [5].

2.1 Population to be modeled

The city of Chicago, Illinois, USA, hosts the largest homeless shelter in the Midwest of the United States, The Pacific Garden Mission (PGM). The spread of COVID-19 at PGM was modeled in [5], based on the contact behavior and population characteristics described below.

The shelter can house up to 950 people. The majority of individuals at PGM, known as overnight residents, spend the night in large dormitories separated by gender, each holding fewer than 200 people. These residents typically spend their days outside the shelter or in large day rooms, also separated by gender. Each night, they return to the same dormitories but with different bed assignments. Prior to the implementation of the stay-at-home order, residents could stay for a maximum of 30 days. A smaller group of residents, referred to as program residents, have a different routine. They sleep in smaller dormitories that range from 4 to 20 beds and can spend their days either in the dormitories, day room, accessing services, or outside the shelter. Depending on the services they are using, these residents can stay at the shelter for up to two years. When the stay-at-home order was issued, more than 50 residents and staff left PGM. Following this order, residents were not allowed to leave or return to the shelter unless they held essential roles such as employment in critical infrastructure. As a result, 445 residents and staff remained at PGM [5].

2.2 Origin of the Outbreak at PGM

On March 14, 2020, a female resident in her 40s from the Pacific Garden Mission (PGM) was diagnosed with COVID-19 at an acute-care hospital. Following this, nine other residents from PGM started showing symptoms and sought medical attention in March. By the end of March, a total of 10 individuals had been confirmed to be infected with SARS-CoV-2 [5].

2.3 The Four-Phase Timeline

For the purposes of the modeling and analysis, the timeline of the COVID-19 outbreak was split into 4 phases [5].

Phase 1 (March 1, 2020 – March 29, 2020): there was no established routine for symptom screening or testing for SARS-CoV-2. If residents reported COVID-19 related symptoms to the staff, they were transported to nearby acute-care hospitals for diagnostic tests and medical treatment [5].

Phase 2 (March 30, 2020 – April 4, 2020): the measures to control infection were extended, and included frequent cleaning of surfaces that are often touched, increasing the supply of hand hygiene products like alcohol-based hand sanitizers, enforcing physical distancing rules, and providing masks to all residents (enough masks for everyone were obtained by April 2). Furthermore, daily temperature checks and symptom screenings were initiated. Residents showing potential COVID-19 symptoms (referred to as persons under investigation [PUIs]) were isolated within the facility. A resident was considered a PUI if they had a measured fever of more than 37.8°C or reported symptoms such as a subjective fever, dry cough, shortness of breath, muscle pain, sore throat, headache, fatigue, or if they had been in close contact with a person confirmed to have SARS-CoV-2 infection. This was in line with the definition provided by the Centers for Disease Control and Prevention (CDC) at that time [5].

Phase 3 (April 5, 2020 – April 7, 2020): persons under investigation (PUIs) were moved to an offsite hotel for isolation in individual rooms. Residents who started showing symptoms were, on average, transferred to the hotel one day after they reported their symptoms. In the meantime, they were isolated onsite. At the same time, residents who were at high risk of severe disease due to their age or underlying health conditions (as determined by an onsite doctor) were also moved offsite to individual hotel rooms for protective housing. A more stringent shelter-in-place order was implemented on April 7, 2020. After this date, residents were strongly advised against leaving the shelter, and those who left for any reason were not allowed to return [5].

Phase 4 (April 8, 2020 – May 13, 2020): was marked by repeated cycles of extensive SARS-CoV-2 testing. From April 8 to 10, healthcare workers from local academic healthcare centers collected nasopharyngeal swab specimens from all staff and residents who consented. The testing was offered to all residents and staff who had not previously tested positive for SARS-CoV-2. The specimens were tested for SARS-CoV-2 using RT-PCR, and related clinical and epidemiological data were gathered using a standardized questionnaire as previously described [12]. Test results were typically returned 48 hours after the collection of the specimen. Onsite isolation units, which were staffed by clinicians 24 hours a day and had the capacity to accommodate 160 individuals, were set up for residents who tested positive for SARS-CoV-2. These isolation units were equipped with a personal protective equipment (PPE) station for medical personnel. Staff and residents received regular training on PPE use, and the PPE station was consistently stocked with surgical and N95 masks, gloves, and gowns. Additional rounds of extensive testing were conducted on April 18, April 28, and May 6. After each round, residents were isolated as previously described. Residents who developed symptoms between testing rounds but did not have an RT-PCR–confirmed diagnosis continued to be transferred to the hotel [5].

2.4 SEIR Model

For many diseases, there is a latent phase during which the individual is infected but not yet infectious. This delay between the acquisition of infection and the infectious state can be incorporated within the SIR model by adding a latent/exposed population, E, and assuming

that Susceptible first enter the latent phase before they become infectious [11]. The resulting **SEIR** model is a compartmental model in epidemiology that is used to describe the spread of an infectious disease. The population is divided into four compartments:

1. **Susceptible** (S): individuals who have not yet been infected but are susceptible to the disease;
2. **Exposed** (E): individuals who have been infected but are not yet infectious themselves. They are in the latency period;
3. **Infectious** (I): individuals who have been infected and are capable of infecting susceptible individuals;
4. **Removed** (R): individuals who have been infected and have either recovered from the disease and are considered permanently immune, or died.

The infectious rate, β, controls the rate of spread, which represents the probability of transmitting the disease between a susceptible and an infectious individual. The rate, σ, of which exposed individuals become infectious (the average duration of incubation period is $\frac{1}{\sigma}$). The recovery rate, $\gamma = \frac{1}{D}$, is determined by the average duration, D, of the infection [11].

The SEIR model is described as a non-linear system of ordinary differential equations. The number of Susceptible, exposed, infectious, and removed change according to

$$\frac{dS}{dt} = -\beta \times S \times I, \tag{2.1}$$

$$\frac{dE}{dt} = \beta \times S \times I - \sigma \times E, \tag{2.2}$$

$$\frac{dI}{dt} = \sigma \times E - \gamma \times I, \tag{2.3}$$

$$\frac{dR}{dt} = \gamma \times I. \tag{2.4}$$

A key parameter in the SEIR model is the basic reproduction number (R_0). It represents the expected number of secondary infections produced by a single infected individual introduced into a population where all other individuals are susceptible, and is calculated as $R_0 = \frac{\beta}{\gamma}$ [11].

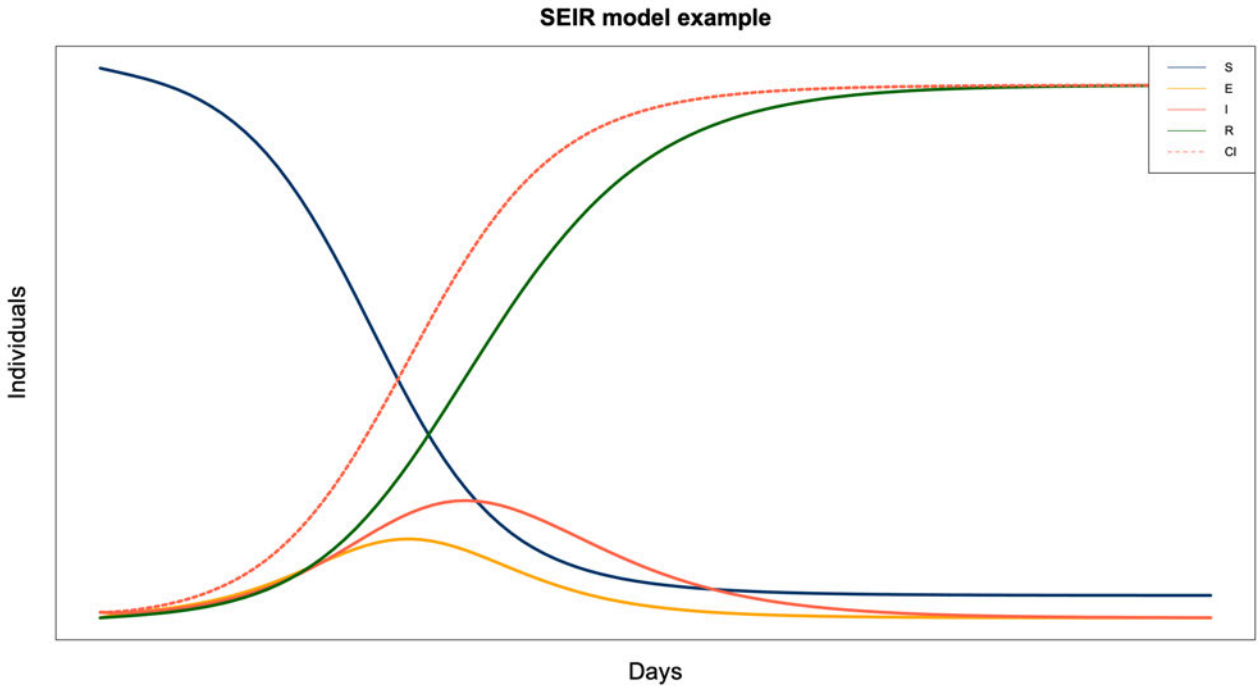

Figure 2.1: An example plot of SEIR model with parameters $\sigma = 0.26$, $\beta = 0.5$, $\gamma = 0.15$ and starting compartments $S = 100$, $E = 0$, $I = 1$, $R = 0$, where S - susceptible, E - exposed, I - infected, R - recovered, CI - cumulative infected.

2.5 Chicago Shelter Model

To understand the dynamics of the PGM outbreak, the SEIR compartmental model was adapted. The constructed model consisted of 4 separate systems of ordinary differential equations corresponding to the 4 phases of outbreak response at PGM [5]. (Figure 2.2)

In each of these phases, the corresponding model parameters and compartments were altered to represent relevant screening, testing, and isolation measures. The model introduced a compartment for isolation units in phase 4 and a compartment for isolation dorms (before the set-up of fully staffed, PPE-stocked isolation units) in phases 2 and 3. Finally, the model included compartments for persons who were removed to the hotel or a hospital [5].

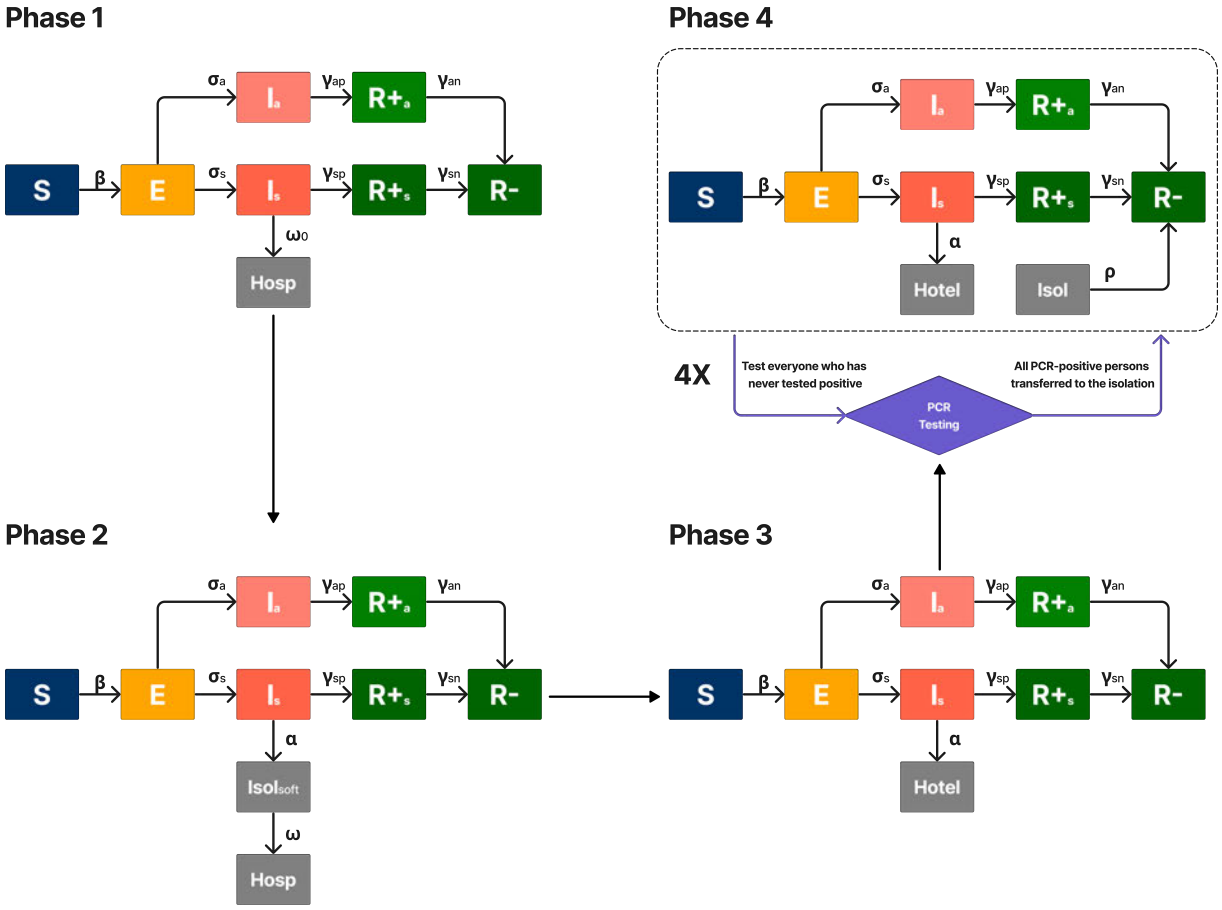

Figure 2.2: Compartmental models corresponding to the 4 phases of the coronavirus disease outbreak response at PGM, a homeless shelter in Chicago, Illinois, USA, 2020.

2.5.1 Transmission Rate

Because transmission rate (β) varies as a function of the number of contacts per infectious person and probability of transmission given contact, it is expected to vary over time in the model because of removal of persons from the population (primarily into isolation units) and infection control measures [5].

Considering that, β at any given time point was calculated by using following the transition equation

$$\beta = \beta_0 - \frac{\beta_0 - \beta_f}{1 + e^{(t - t_{\text{Trans}})/k}}, \tag{2.5}$$

where β_0 corresponds to the initial transmission rate, $\beta_f = \beta_f\_pct\_\beta_0 \times \beta_0$ (where $\beta_f\_pct\_\beta_0$ corresponds to final transmission rate as a percentage of β_0), t_{Trans} represents the time point at which β reaches a value halfway between β_0 and β_f, and k represents the rate of transformation between initial and final β [5].

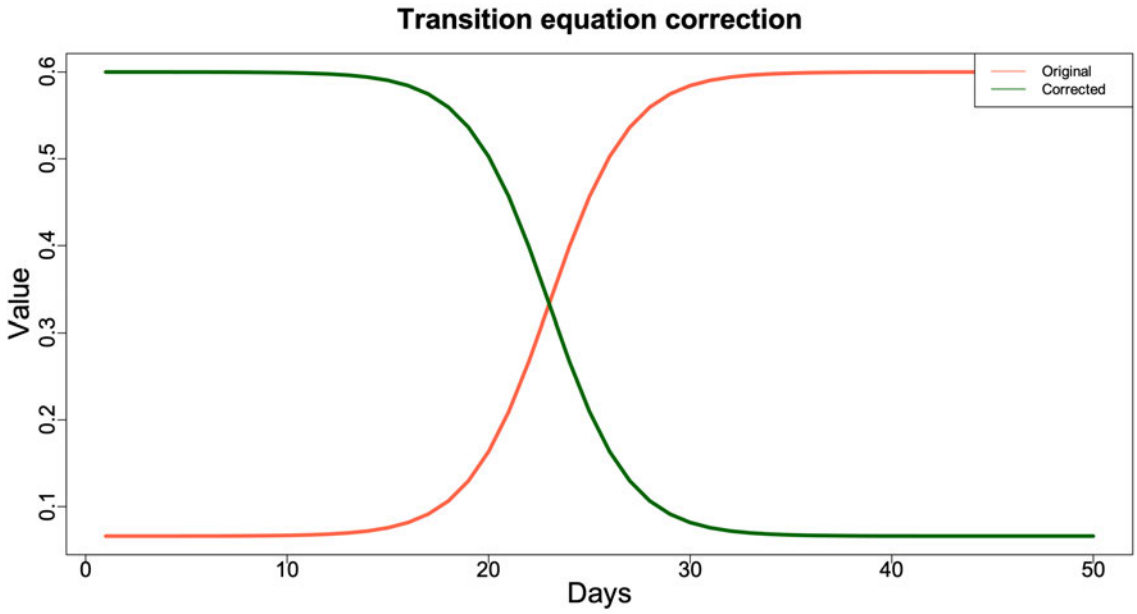

Figure 2.3: Comparison of the original β function with the corrected one. Parameters values are $\beta_0 = 0.6$, $\beta_f = 0.066$, $t_{Trans} = 23$ and $k = 2$.

However, the original transition equation produces a function increasing with time, which contradicts the expectation stated (Figure 2.3). To correct the situation, the equation was changed to

$$\beta = \beta_0 - \frac{\beta_0 - \beta_f}{1 + e^{(t_{Trans} - t)/k}}. \tag{2.6}$$

2.5.2 Differential Equations

The ordinary differential equations (ODEs) from the "Transmission Dynamics of Large Coronavirus Disease Outbreak in Homeless Shelter, Chicago, Illinois, USA, 2020" [5] had some inaccuracies, which are corrected in the ODEs below. Compartments, variables, and their descriptions are provided in the Table 2.1.

Notation	Description
S	Susceptible persons
E	Exposed persons
I_{s}	Infectious symptomatic persons
I_{a}	Infectious asymptomatic persons
R_{ps}	Recovered symptomatic persons, PCR-positive
R_{pa}	Recovered asymptomatic persons, PCR-positive
R_{n}	Recovered persons, PCR-negative
H	Hospitalized persons (phase 1, phase 2)
Q	Soft-isolated persons (phase 2)
Ht	Persons suspected of having COVID-19 and transferred to a hospital (phase 3, phase 4)
$Isol$	Isolated as a result of 4 rounds of PCR-testing (phase 4)
β	Rate of transmission between Susceptible and Infectious persons
σ_{s}	Rate of transition from E to I_{s} $\sigma_{\text{s}} = 1/t_{\text{incubation}} \times p_{\text{symp}}$, where $t_{\text{incubation}}$ is incubation period and p_{symp} is percent symptomatic
σ_{a}	Rate of transition from E to I_{a} $\sigma_{\text{a}} = 1/t_{\text{incubation}} \times p_{\text{asymp}}$, where $t_{\text{incubation}}$ is incubation period and p_{asymp} is percent asymptomatic
γ_{sp}	Rate of transition from I_{s} to R_{ps} $\gamma_{\text{sp}} = 1/t_{\text{infectious\_s}}$, where $t_{\text{infectious\_s}}$ is an infectious period of symptomatic persons
γ_{ap}	Rate of transition from I_{a} to R_{pa} $\gamma_{\text{ap}} = 1/t_{\text{infectious\_a}}$, where $t_{\text{infectious\_a}}$ is an infectious period of asymptomatic persons
γ_{sn}	Rate of transition from R_{ps} to R_{n} $\gamma_{\text{sn}} = 1/(t_{\text{pcrPos\_s}} - t_{\text{infectious\_s}})$, where $t_{\text{pcrPos\_s}}$ is duration of PCR-positivity for symptomatic infected persons
γ_{an}	Rate of transition from R_{pa} to R_{n} $\gamma_{\text{an}} = 1/(t_{\text{pcrPos\_a}} - t_{\text{infectious\_a}})$, where $t_{\text{pcrPos\_a}}$ is duration of PCR-positivity for asymptomatic infected persons
ω_0	Rate of hospital admission of I_{s} (Phase 1)
ω	Rate of transition from Q to H (Phase 2)
α	Rate of transition from I_{s} to Q (Phase 2); rate of transition from I_{s} to Ht (Phase 3, Phase 4)
ρ	Rate of transition from $Isol$ to R_{n} (Phase 4); $\rho = 1/t_{\text{isolation}}$, where $t_{\text{isolation}}$ is duration of isolation (14 days)

Table 2.1: Compartments and variables of ODEs for phases 1-4.

The following system of ODEs describes **phase 1**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.7}$$
$$\frac{dE}{dt} = \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E - \sigma_a \times E, \tag{2.8}$$
$$\frac{dI_s}{dt} = \sigma_s \times E - \gamma_{sp} \times I_s - \omega_0 \times I_s, \tag{2.9}$$
$$\frac{dI_a}{dt} = \sigma_a \times E - \gamma_{ap} \times I_a, \tag{2.10}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.11}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.12}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.13}$$
$$\frac{dH}{dt} = \omega_0 \times I_s. \tag{2.14}$$

In phase 2 the isolation dorms (Q) were introduced, due to the commencement of symptom screening. These are the ODEs for **phase 2**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.15}$$
$$\frac{dE}{dt} = \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E - \sigma_a \times E, \tag{2.16}$$
$$\frac{dI_s}{dt} = \sigma_s \times E - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.17}$$
$$\frac{dI_a}{dt} = \sigma_a \times E - \gamma_{ap} \times I_a, \tag{2.18}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.19}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.20}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.21}$$
$$\frac{dQ}{dt} = -\omega \times Q + \alpha \times I_s, \tag{2.22}$$
$$\frac{dH}{dt} = \omega \times Q. \tag{2.23}$$

In phase 3 hospital (H) and isolation dorms (Q) compartments were replaced with Hotel (Ht) due to the opening of a hotel for homeless persons suspected to have COVID-19. All symptomatic persons were moved to the hotel once tested positive. These are ODEs for **phase 3**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.24}$$
$$\frac{dE}{dt} = \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E - \sigma_a \times E, \tag{2.25}$$
$$\frac{dI_s}{dt} = \sigma_s \times E - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.26}$$
$$\frac{dI_a}{dt} = \sigma_a \times E - \gamma_{ap} \times I_a, \tag{2.27}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.28}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.29}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.30}$$
$$\frac{dHt}{dt} = \alpha \times I_s. \tag{2.31}$$

In phase 4 the isolation units for persons who tested positive during mass PCR screens were implemented. At each of the 4 isolation time points (2 days after each testing), the number of persons in the I_s, I_a, R_{ps}, and R_{pa} compartments who are simulated to test positive (SensitivityPCR $\times$ number_of_individuals_in_each_compartment) are moved to the Isolation compartment (Q). These are ODEs for **phase 4**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.32}$$
$$\frac{dE}{dt} = \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E - \sigma_a \times E, \tag{2.33}$$
$$\frac{dI_s}{dt} = \sigma_s \times E - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.34}$$
$$\frac{dI_a}{dt} = \sigma_a \times E - \gamma_{ap} \times I_a, \tag{2.35}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.36}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.37}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa} + \rho \times Isol, \tag{2.38}$$
$$\frac{dHt}{dt} = \alpha \times I_s, \tag{2.39}$$
$$\frac{dIsol}{dt} = -\rho \times Isol. \tag{2.40}$$

2.6 Alternative Model

Although the incubation period is the same for symptomatic and asymptomatic individuals, the model has to parametrize σ_s and σ_a, which have to be carefully interpreted. By extending the model with an additional compartment, more straightforward interpretation of parameters is achieved and the assumption that incubation period is the same for symptomatic and asymptomatic individuals, can be relaxed. More precisely, we split the E compartment into two parts: E_a and E_s, which represent exposed individuals that later become asymptomatic and symptomatic infected, respectively. In this variation of the model, the rate of transition from Susceptible to E_a is $\beta \times p$, where p is a fraction of asymptomatic infectious, or a probability of a person being asymptomatic. The rate of transition from S compartment to E_s in this case is $\beta \times (1 - p)$. After transition to either exposed compartments, a person can be transitioned further only to a corresponding infectious compartment. Rates of transition from exposed to I_a and I_s, unlike the base model, are not affected by p. Those rates are

$$\sigma_s = \frac{1}{t_{\text{incubation\_symptomatic}}}, \tag{2.41}$$

$$\sigma_a = \frac{1}{t_{\text{incubation\_asymptomatic}}}, \tag{2.42}$$

where $t_{\text{incubation\_symptomatic}}$ is incubation period for symptomatic persons, and $t_{\text{incubation\_asymptomatic}}$ is incubation period for asymptomatic persons. The alternative model of phase 1 is illustrated in Figure 2.4.

The alternative model was implemented for every phase of the PGM outbreak alongside the base model.

The ODEs for the alternative model are similar to the ones from the base model, with difference only in exposed and infected compartments.

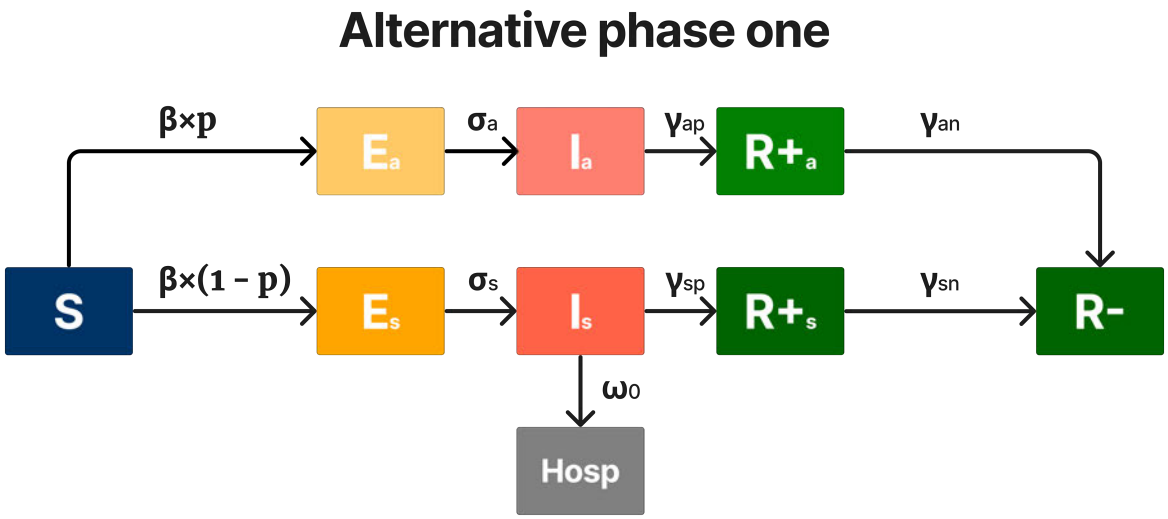

Figure 2.4: Phase one compartments of the alternative model.

The following system of ODEs describes **phase 1**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.43}$$
$$\frac{dE_s}{dt} = (1-p) \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E_s, \tag{2.44}$$
$$\frac{dE_a}{dt} = p \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_a \times E_a, \tag{2.45}$$
$$\frac{dI_s}{dt} = \sigma_s \times E_s - \gamma_{sp} \times I_s - \omega_0 \times I_s, \tag{2.46}$$
$$\frac{dI_a}{dt} = \sigma_a \times E_a - \gamma_{ap} \times I_a, \tag{2.47}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.48}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.49}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.50}$$
$$\frac{dH}{dt} = \omega_0 \times I_s. \tag{2.51}$$

The following system of ODEs describes **phase 2**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.52}$$
$$\frac{dE_s}{dt} = (1-p) \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E_s, \tag{2.53}$$
$$\frac{dE_a}{dt} = p \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_a \times E_a, \tag{2.54}$$
$$\frac{dI_s}{dt} = \sigma_s \times E_s - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.55}$$
$$\frac{dI_a}{dt} = \sigma_a \times E_a - \gamma_{ap} \times I_a, \tag{2.56}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.57}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.58}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.59}$$
$$\frac{dQ}{dt} = -\omega \times Q + \alpha \times I_s, \tag{2.60}$$
$$\frac{dH}{dt} = \omega \times Q. \tag{2.61}$$

The following system of ODEs describes **phase 3**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.62}$$
$$\frac{dE_s}{dt} = (1-p) \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E_s, \tag{2.63}$$
$$\frac{dE_a}{dt} = p \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_a \times E_a, \tag{2.64}$$
$$\frac{dI_s}{dt} = \sigma_s \times E_s - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.65}$$
$$\frac{dI_a}{dt} = \sigma_a \times E_a - \gamma_{ap} \times I_a, \tag{2.66}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.67}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.68}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa}, \tag{2.69}$$
$$\frac{dHt}{dt} = \alpha \times I_s. \tag{2.70}$$

The following system of ODEs describes **phase 4**:

$$\frac{dS}{dt} = -\beta \times S \times \frac{(I_s + I_a)}{N}, \tag{2.71}$$
$$\frac{dE_s}{dt} = (1-p) \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_s \times E_s, \tag{2.72}$$
$$\frac{dE_a}{dt} = p \times \beta \times S \times \frac{(I_s + I_a)}{N} - \sigma_a \times E_a, \tag{2.73}$$
$$\frac{dI_s}{dt} = \sigma_s \times E_s - \gamma_{sp} \times I_s - \alpha \times I_s, \tag{2.74}$$
$$\frac{dI_a}{dt} = \sigma_a \times E_a - \gamma_{ap} \times I_a, \tag{2.75}$$
$$\frac{dR_{ps}}{dt} = \gamma_{sp} \times I_s - \gamma_{sn} \times R_{ps}, \tag{2.76}$$
$$\frac{dR_{pa}}{dt} = \gamma_{ap} \times I_a - \gamma_{an} \times R_{pa}, \tag{2.77}$$
$$\frac{dR_n}{dt} = \gamma_{sn} \times R_{ps} + \gamma_{an} \times R_{pa} + \rho \times Isol, \tag{2.78}$$
$$\frac{dHt}{dt} = \alpha \times I_s, \tag{2.79}$$
$$\frac{dIsol}{dt} = -\rho \times Isol. \tag{2.80}$$

2.7 Model Fitting

To fit model parameters, two functions to propagate all 4 model phases sequentially were constructed, for the base and alternative model. Onward, the $optim$ function from $stats$ package of R [19] was used with the limited memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) optimization algorithm. The L-BFGS is an optimization algorithm in the family of quasi-Newton methods that approximates the Broyden–Fletcher–Goldfarb–Shanno algorithm (BFGS) using a limited amount of computer memory [13].

Table 2.2 presents the data points, values, and weights that were used to minimize the root mean log squared error during the model fitting process.

Table 2.2: Data points for model fitting of coronavirus disease outbreak in PGM.

Description	Phase	Value	Weight
Cum. number of hospital-based PCR+, March 14	I	1	1
Cum. number of hospital-based PCR+, March 15	I	1	1
Cum. number of hospital-based PCR+, March 16	I	1	1
Cum. number of hospital-based PCR+, March 17	I	2	1
Cum. number of hospital-based PCR+, March 18	I	3	1
Cum. number of hospital-based PCR+, March 19	I	3	1
Cum. number of hospital-based PCR+, March 20	I	4	1
Cum. number of hospital-based PCR+, March 21	I	4	1
Cum. number of hospital-based PCR+, March 22	I	5	1
Cum. number of hospital-based PCR+, March 23	I	6	1
Cum. number of hospital-based PCR+, March 24	I	6	1
Cum. number of hospital-based PCR+, March 25	I	6	1
Cum. number of hospital-based PCR+, March 26	I	7	1
Cum. number of hospital-based PCR+, March 27	I	9	1
Cum. number of hospital-based PCR+, March 28	I	9	1
Cum. number of hospital-based PCR+, March 29	I	9	1
Cum. number of hospital-based PCR+, March 30	I	10	1
Number of PCR+ persons	II	18	1
Number of hospitalized persons	II	7	1
Number of persons moved to the hotel	III	26	1
Number of PCR+ in the first round of mass PCR testing	IV	166	2
Number of PCR+ in the second round of mass PCR testing	IV	24	2
Number of PCR+ in the third round of mass PCR testing	IV	23	2
Number of PCR+ in the fourth round of mass PCR testing	IV	1	2
Number of persons moved to the hotel between 1 and 2 rounds of mass PCR testing	IV	20	1
Number of persons moved to the hotel between 2 and 3 rounds of mass PCR testing	IV	4	1
Number of persons moved to the hotel between 3 and 4 rounds of mass PCR testing	IV	0	1

Variable	Range of values	Phases
β_0 — initial β	0-445	1, 2, 3, 4
β_f_pct_β_0 - final β as a percentage of β_0	0-1	1, 2, 3, 4
k — rate of transformation of β	0.01–2	1, 2, 3, 4
t_{Trans} — day when β reaches halfway between β_0 and β_f	1-50	1, 2, 3, 4
$t_{incubation}$ — time between E and I compartments	2.8-4.0	1, 2, 3, 4
p — asymptomatic percentage	0.18–0.87	1, 2, 3, 4
$t_{infectious\_s}$ — infectious duration for symptomatic persons (days)	3-8	1, 2, 3, 4
$t_{infectious\_a}$ — infectious duration for asymptomatic persons (days)	3-8	1, 2, 3, 4
t_{pcrPos_s} — duration of RT-PCR–positivity of symptomatic persons (days)	16-35	1, 2, 3, 4
t_{pcrPos_a} — duration of RT-PCR–positivity of asymptomatic persons (days)	3-35	1, 2, 3, 4
α — rate of detection of symptomatic infectious persons through screening	0.01-1	2, 3, 4
$rt\_pcr\_sensitivity$ — RT-PCR sensitivity	0.72–0.90	4
ω_0 — rate of hospital admission of Infectious symptomatic persons before screening	0.05–1.0	1
ω — rate of hospital admission of soft-isolated symptomatic persons	0.05–1.0	2

Table 2.3: Model parameters' ranges for fitting.

Ranges of values for each optimized variable (Table 2.3) were derived from the literature[5]

The basic reproduction number (R_0), which is calculated as β/γ in a basic SEIR model, was calculated as $\beta_0/(\gamma_{ap} \times p + \gamma_{sp} \times (1-p))$, where γ_{ap} is the inverse of infectious duration for asymptomatic persons and γ_{sp} is the inverse of infectious duration for symptomatic persons. The number of persons in different compartments at various time points and model parameters (representing transmission dynamics) were estimated from the fitted model [5].

3 Results

Here, we present the results of the model simulations.

3.1 Base model

The $optim$ method with L-BFGS algorithm converged, giving the following results (Table 3.1).

Based on the fitted variables, following derivative parameters were calculated (Table 3.2), according to equations from Table 2.1.

Fitted value	Result	Description
β_0	1	Initial β
β_f_pct_β_0	0.22	final β as a percentage of β_0
t_{Trans}	28.73	Moment where β reaches halfway between β_0 and β_f (days)
k	0.01	Rate of transformation of β
p	0.36	Asymptomatic percentage
$t_{\text{incubation}}$	2.8	Time between E and I compartments (days)
$t_{\text{infectious\_s}}$	5.12	infectious duration for symptomatic persons (days)
$t_{\text{infectious\_a}}$	8	Infectious duration for asymptomatic persons (days)
$t_{\text{pcrPos\_s}}$	35	Duration of RT-PCR–positivity of symptomatic persons (days)
$t_{\text{pcrPos\_a}}$	35	Duration of RT-PCR–positivity of asymptomatic persons (days)
ω_0	0.86	Rate of hospital admission of infectious symptomatic persons before screening
α	0.29	Rate of detection of symptomatic infectious persons through screening
ω	0.7	rate of hospital admission of soft-isolated symptomatic persons during phase 2
$rt\_pcr\_sensitivity$	0.9	RT-PCR sensitivity

Table 3.1: Fitted parameters for the base model

Parameter	Result	Description
σ_s	0.23	Rate of transition from E to I_s
σ_a	0.13	Rate of transition from E to I_a
γ_{sp}	0.2	Rate of transition from I_s to R_{ps}
γ_{ap}	0.13	Rate of transition from I_a to R_{pa}
γ_{sn}	0.03	Rate of transition from R_{ps} to R_n
γ_{an}	0.04	Rate of transition from R_{pa} to R_n

Table 3.2: Derivative model parameters calculated based on fitted model parameters from Table 3.1

The trajectories of the compartmental model, which include susceptible, exposed, infectious, recovered, and additionally total infected individuals over time, are shown in Figure 3.1.

The results show that the spread of infection was slowed down considerably with the start of mass-testing in phase 4.

The model gave R_0 value of 5.78, which suggest that one person with COVID-19 could infect 5–6 people. According to the modeling, a total of 276 people were infected at some point during the observation.

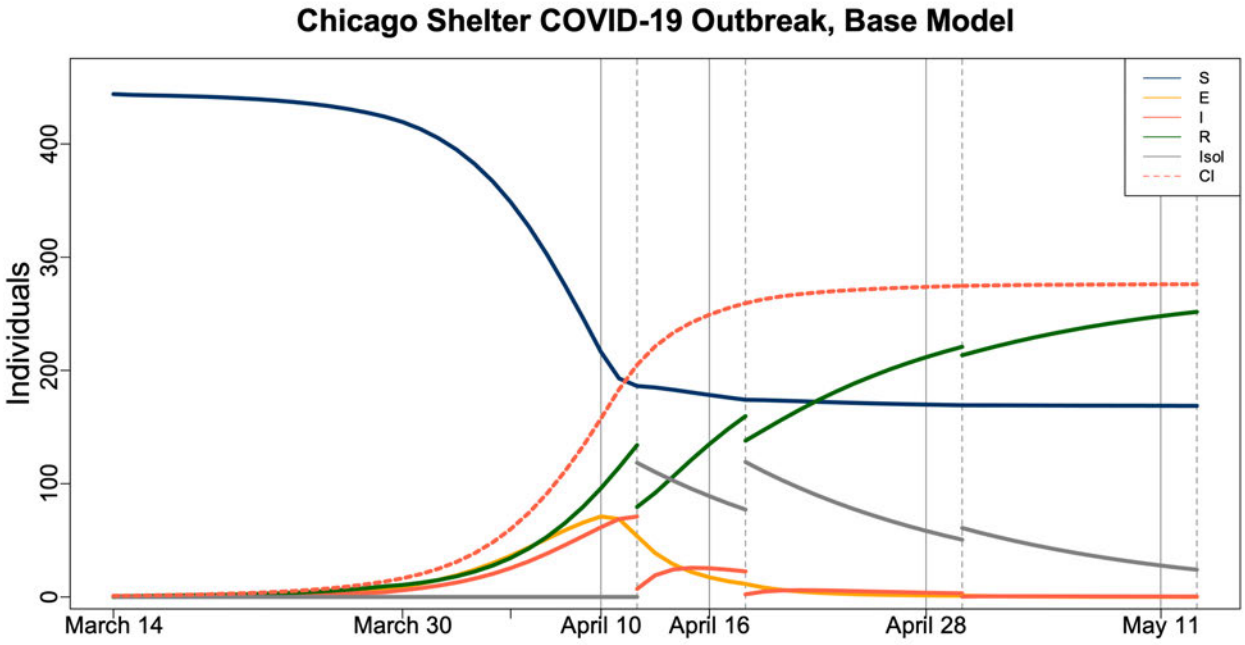

Figure 3.1: Compartmental modeling results of the coronavirus disease outbreak at Pacific Garden Mission, a homeless shelter in Chicago, Illinois, USA, 2020. Time points corresponding to each of the 4 rounds of mass testing and isolation are indicated by vertical dotted lines and vertical dashed lines. The susceptible compartment corresponds to persons who are estimated to have never been infected; exposed persons have been infected but are not yet infectious; infectious includes persons in both I_s and I_a; recovered include the R_{ps}, R_{pa}, and R_n compartments; $Isol$ compartment refers to persons isolated as a result of mass-testing rounds in phase 4; CI refers to cumulative infected persons.

3.2 Alternative model

For the alternative model (2.43), the L-BFGS algorithm yielded the following results (Table 3.3). The difference was, as mentioned previously, that the incubation period for symptomatic and asymptomatic persons was fitted as two different parameters.

Based on the fitted variables, following derivative parameters were calculated (Table 3.4).

Fitted value	Result	Description
β_0	1	Initial $beta$
β_f_pct_β_0	0.08	final β as a percentage of β_0
t_{Trans}	29.15	Moment where β reaches halfway between β_0 and β_f (days)
k	0.01	Rate of transformation of β
p	0.45	Asymptomatic percentage
$t_{incubation\_a}$	4	Time between E_a and I_a compartments (days)
$t_{incubation\_s}$	2.8	Time between E_s and I_s compartments (days)
$t_{infectious\_s}$	8	infectious duration for symptomatic persons (days)
$t_{infectious\_a}$	8	Infectious duration for asymptomatic persons (days)
t_{pcrPos_s}	35	Duration of RT-PCR–positivity of symptomatic persons (days)
t_{pcrPos_a}	35	Duration of RT-PCR–positivity of asymptomatic persons (days)
ω_0	0.84	Rate of hospital admission of infectious symptomatic persons before screening
α	0.38	Rate of detection of symptomatic infectious persons through screening
ω	0.56	rate of hospital admission of soft-isolated symptomatic persons during phase 2
$rt\_pcr\_sensitivity$	0.9	RT-PCR sensitivity

Table 3.3: Fitted model parameters for the alternative model

Parameter	Result	Description
σ_s	0.36	Rate of transition from E_s to I_s
σ_a	0.25	Rate of transition from E_a to I_a
γ_{sp}	0.13	Rate of transition from I_s to R_{ps}
γ_{ap}	0.13	Rate of transition from I_a to R_{pa}
γ_{sn}	0.04	Rate of transition from R_{ps} to R_n
γ_{an}	0.04	Rate of transition from R_{pa} to R_n

Table 3.4: Derivative model parameters calculated based on fitted model parameters from Table 3.3

The trajectories of the compartmental model, which include susceptible, exposed, infectious, recovered, and additionally total infected individuals over time, are shown in Figure 3.2.

As in the base model, the results for the alternative model show that the spread of infection was slowed down considerably with the start of mass-testing in phase 4.

While the results are not surprisingly similar, it is important to note that the estimated incubation period for symptomatic and asymptomatic infectious differ.

The model gave R_0 value of 8, which is considerably greater than the R_0 of the base model. According to the modeling, a total of 271 people were infected at some point during the observation.

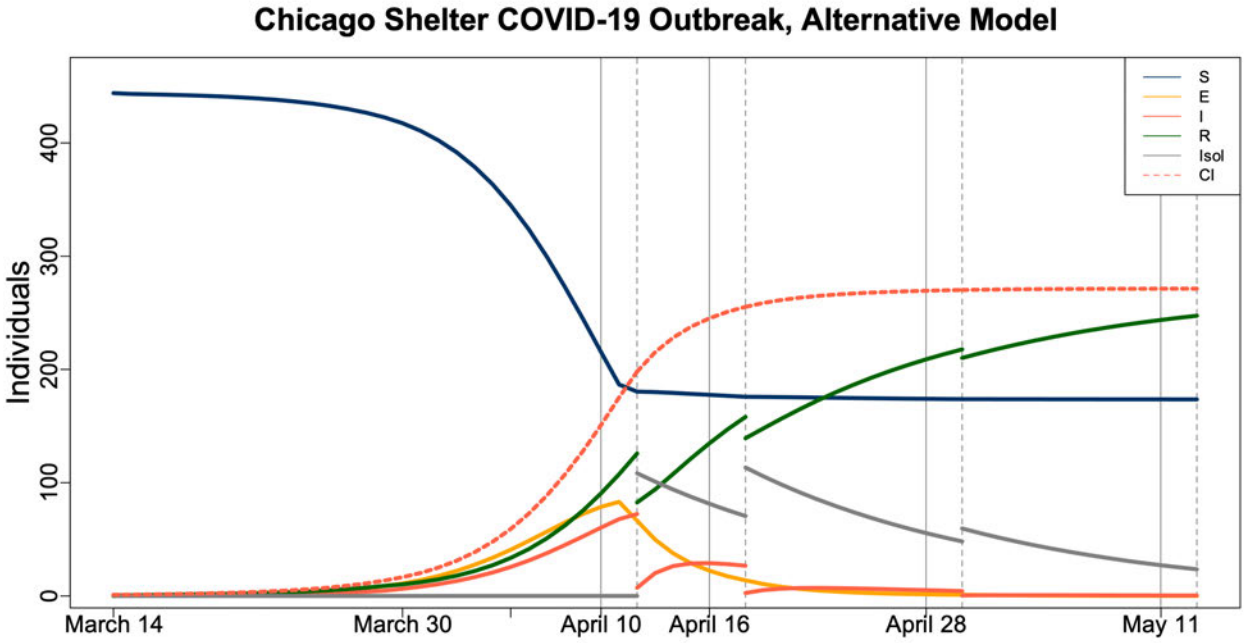

Figure 3.2: Compartmental modeling results with the alternative approach of the coronavirus disease outbreak at Pacific Garden Mission, a homeless shelter in Chicago, Illinois, USA, 2020. Time points corresponding to each of the 4 rounds of mass testing and isolation are indicated by vertical dotted lines and vertical dashed lines. The susceptible compartment corresponds to persons who are estimated to have never been infected; exposed persons have been infected but are not yet infectious; infectious includes persons in both I_s and I_a; recovered include the R_{ps}, R_{pa}, and R_n compartments; $Isol$ compartment refers to persons isolated as a result of mass-testing rounds in phase 4; CI refers to cumulative infected persons

4 Discussion

Constructing such models allows discovering which of the virus control methods are most effective, and it is crucially important in planning response to the future threats similar to COVID-19.

In this study, we corrected and implemented a model of COVID-19 outbreak at PGM homeless shelter, Chicago [5]. Additionally, an alternative approach based on fewer assumptions was demonstrated, which gave evidence that the incubation periods for symptomatic and asymptomatic individuals can differ.

4.1 Comparison of implementations

Aside from the corrected transmission rate function mentioned in (2.5) and other minor misprints, our implementation differs from the original implementation in [5] in usage of an additional transmission rate parameter in phases 2 (λ_0) and 4 (λ). Their presence in the original implementation was explained by the addition of the isolation compartments ($Isol_{\text{soft}}$ and $Isol$). It was unclear how the addition of such compartments could increase the speed of transition from S to E, and it contradicted the idea of the gradually decreasing transmission rate over time due to the measures taken. Furthermore, the model with λ_0 and λ was implemented with the fitting ranges specified in [5], but the fitted values were 0, which effectively confirmed their redundancy.

The basic reproduction number yielded by our implementation of the model is 5.78, which is not only greater than early estimates (2.2 – 2.7[20]), but also greater than the result of the original implementation (4.5) [5]. However, other studies indicate R_0 value extremely close to the value resulted by our model [20].

The original implementation of the model indicated that approximately 350 persons were cumulatively infected, which was significantly more than 253 cases detected by PCR testing [5]. At the same time, the model in both implementations produces the PCR-sensitivity of 90%, which gives an approximation of 281 total infected, which is much closer to our value of 276 cumulative infected persons.

Another significant difference in the results could be seen in the value of p, the asymptomatic percentage. The original implementation of the model yielded $p = 72\%$, whereas our yielded $p = 36\%$ [5]. More recent studies demonstrate value closer to 40% [15].

4.2 Base and alternative models

As could be seen from figures 3.1 and 3.2, the alternative model yielded roughly the same values for the compartments and cumulative infected persons. However, the fitted results for the incubation rates for symptomatic and asymptomatic individuals are different, which could not be reflected in the base model.

Additionally, the alternative model is far more intuitive. In the base model, incubation period can not be calculated as $1/\sigma_s$ or $1/\sigma_a$, which could be expected. The model also implies that the incubation period is the same for symptomatic and asymptomatic individuals, although it has two different parameters for incubation rate.

4.3 Limitations

The reported estimates, such as the duration of viral shedding, show high variance in the population and may not be normally distributed [4]. One study of 21 patients with mild symptoms found that 90% of them had repeated negative RT-PCR tests within 10 days of symptom onset [14]. Another study of 56 patients with mild to moderate symptoms reported a median viral RNA shedding duration of 24 days [1].

The underlying test data were limited by the lack of widespread testing. Widespread testing in congregate settings was not established in Chicago until April 2020, and no widespread testing data were available to characterize the first phase of this outbreak. Our model accounts for this early lack of testing and fits compartmental trajectories across the entire outbreak timeline. It uses known ranges for parameters like infectious duration and RT-PCR-positive duration, but it inevitably simplifies some aspects of the context [5].

This simplification, along with the large number of fitted parameters, necessitates careful interpretation of the fitted parameter values. Other limitations include the assumption of a closed system. Although the shelter did not allow residents to enter or leave, some high-risk residents were preemptively moved to a hotel, and some residents inevitably left the shelter. Additionally, some staff members left the shelter and returned. The model also assumes random mixing of the shelter population outside of isolation units, which was not fully the case due to the factors such as gender-separated rooms [5].

Finally, the parameter uncertainties and the small population size might indicate the necessity of an implementation of a stochastic or a hybrid model for achieving better results. Additionally, sample size is too small to give confidence in the results, such as different incubation periods for symptomatic and asymptomatic individuals in the alternative model.

Appendix A: Common

A.1 Plots

Listing A.1: plot.R

```r
red <- "#FF6347"
green <- "#006400"
blue <- "#003366"
orange <- "#FFA500"
gray <- "#808080"

lines_plot <- "l"
solid_line <- 1
dashed_line <- 2
line_thickness <- 4
bottom_side <- 1

axis_values_size <- 1.5
axis_labels_size <- 2
title_size <- 2
axis_title_margin <- 2
axis_label_margin <- 0.5
axis_line_margin <- 0

par(cex.main = title_size)
par(cex.axis = axis_values_size)
par(cex.lab = axis_labels_size)
par(mgp = c(axis_title_margin, axis_label_margin, axis_line_margin))

beta_plot <- function(df, header) {
  number_of_columns <- length(df)
  column_names <- colnames(df)[1:number_of_columns]
  column_types <- rep(solid_line, number_of_columns)
  column_colors <- c(red, green)[1:number_of_columns]

  matplot(
    x = 1:50,
    y = df,
    type = "l",
    main = header,
    xlab = "Days",
    ylab = "Value",
    col = column_colors,
    lty = solid_line,
```

```
    lwd = line_thickness
  )
  legend("topright", legend=column_names, col=column_colors, lty =
      column_types)

}

seir_plot <- function(df, header, starting_date = as.Date("2000-01-01")
    ) {
  ci_present <- ("CI" %in% names(df))
  if (ci_present) {
    ci <- df$CI
    df$CI <- NULL
  }
  df$time <- NULL

  number_of_columns <- length(df)
  number_of_rows <- nrow(df)
  column_names <- colnames(df)[1:number_of_columns]
  column_types <- rep(solid_line, number_of_columns)
  column_colors <- c(blue, orange, red, green, gray, red)[1:number_of_
      columns]

  date_sequence <- seq(starting_date, by = "days", length.out = number_
      of_rows)
  matplot(
    x = date_sequence,
    y = df,
    type = lines_plot,
    lty = solid_line,
    main = header,
    xlab = "Days",
    ylab = "Individuals",
    col = column_colors,
    xaxt = if (starting_date == as.Date("2000-01-01")) "n" else "s",
    yaxt = "n",
    lwd = line_thickness
  )

  if (ci_present) {
    lines(date_sequence, ci, type=lines_plot, lty=dashed_line, col=red,
        lwd = line_thickness)
    column_names <- c(column_names, "CI")
    column_types <- c(column_types, dashed_line)
    column_colors <- c(column_colors, red)
  }

```

```
  legend("topright", legend=column_names, col=column_colors, lty =
      column_types)
}

shelter_plot <- function(df, header) {
  df$time <- NULL

  number_of_columns <- length(df)
  column_names <- colnames(df)[1:number_of_columns]
  column_types <- c(rep(solid_line, number_of_columns - 1), dashed_line
      )
  column_colors <- c(blue, orange, red, green, gray, red)[1:number_of_
      columns]
  observation_duration_days <- 61
  date_sequence <- seq(as.Date("2020-03-14"), by = "days", length.out =
       observation_duration_days)

  # Creating a plot with an invisible line to set the correct scale
  plot(
    x = date_sequence,
    y = 7.5 * seq(1, observation_duration_days),
    type = lines_plot,
    lty = 0,
    main = header,
    xlab = "",
    ylab = "Individuals",
    col=rgb(0, 0, 0, alpha = 0.0),
    xaxt = "n"
  )

  # Displaying key dates
  dates_to_show <- c(
    as.Date("2020-03-14"),
    as.Date("2020-03-30"),
    as.Date("2020-04-05"),
    as.Date("2020-04-10"),
    as.Date("2020-04-16"),
    as.Date("2020-04-28"),
    as.Date("2020-05-11")
  )
  axis(side = bottom_side, at = dates_to_show, labels =  format(dates_
      to_show, "%B %d"))

  # Marking 4 rounds of mass-testing in phase 4
  abline(v = as.Date("2020-04-10"), col = rgb(0, 0, 0, alpha = 0.5),
      lty = solid_line)
```

```
122   abline(v = as.Date("2020-04-12"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = dashed_line)
123
124   abline(v = as.Date("2020-04-16"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = solid_line)
125   abline(v = as.Date("2020-04-18"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = dashed_line)
126
127   abline(v = as.Date("2020-04-28"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = solid_line)
128   abline(v = as.Date("2020-04-30"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = dashed_line)
129
130   abline(v = as.Date("2020-05-11"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = solid_line)
131   abline(v = as.Date("2020-05-13"), col = rgb(0, 0, 0, alpha = 0.5),
          lty = dashed_line)
132
133   # Creating the lines by phases to get the correct connections on the
          plot
134   part_one <- df[1:30, ]
135   part_one_dates <- date_sequence[1:30]
136   lines(part_one_dates, part_one$S, type=lines_plot, lty=solid_line,
          col=blue, lwd = line_thickness)
137   lines(part_one_dates, part_one$E, type=lines_plot, lty=solid_line,
          col=orange, lwd = line_thickness)
138   lines(part_one_dates, part_one$I, type=lines_plot, lty=solid_line,
          col=red, lwd = line_thickness)
139   lines(part_one_dates, part_one$R, type=lines_plot, lty=solid_line,
          col=green, lwd = line_thickness)
140   lines(part_one_dates, part_one$Isol, type=lines_plot, lty=solid_line,
           col=gray, lwd = line_thickness)
141   lines(part_one_dates, part_one$CI, type=lines_plot, lty=dashed_line,
          col=red, lwd = line_thickness)
142
143
144   # part two
145   part_two <- df[31:37, ]
146   part_two_dates <- date_sequence[30:36]
147   lines(part_two_dates, part_two$S, type=lines_plot, lty=solid_line,
          col=blue, lwd = line_thickness)
148   lines(part_two_dates, part_two$E, type=lines_plot, lty=solid_line,
          col=orange, lwd = line_thickness)
149   lines(part_two_dates, part_two$I, type=lines_plot, lty=solid_line,
          col=red, lwd = line_thickness)
150   lines(part_two_dates, part_two$R, type=lines_plot, lty=solid_line,
          col=green, lwd = line_thickness)
```

```
  lines(part_two_dates, part_two$Isol, type=lines_plot, lty=solid_line,
      col=gray, lwd = line_thickness)
  lines(part_two_dates, part_two$CI, type=lines_plot, lty=dashed_line,
     col=red, lwd = line_thickness)

  # part three
  part_three <- df[38:50, ]
  part_three_dates <- date_sequence[36:48]
  lines(part_three_dates, part_three$S, type=lines_plot, lty=solid_line
     , col=blue, lwd = line_thickness)
  lines(part_three_dates, part_three$E, type=lines_plot, lty=solid_line
     , col=orange, lwd = line_thickness)
  lines(part_three_dates, part_three$I, type=lines_plot, lty=solid_line
     , col=red, lwd = line_thickness)
  lines(part_three_dates, part_three$R, type=lines_plot, lty=solid_line
     , col=green, lwd = line_thickness)
  lines(part_three_dates, part_three$Isol, type=lines_plot, lty=solid_
     line, col=gray, lwd = line_thickness)
  lines(part_three_dates, part_three$CI, type=lines_plot, lty=dashed_
     line, col=red, lwd = line_thickness)

  # part four
  part_four <- df[51:64, ]
  part_four_dates <- date_sequence[48:61]
  lines(part_four_dates, part_four$S, type=lines_plot, lty=solid_line,
     col=blue, lwd = line_thickness)
  lines(part_four_dates, part_four$E, type=lines_plot, lty=solid_line,
     col=orange, lwd = line_thickness)
  lines(part_four_dates, part_four$I, type=lines_plot, lty=solid_line,
     col=red, lwd = line_thickness)
  lines(part_four_dates, part_four$R, type=lines_plot, lty=solid_line,
     col=green, lwd = line_thickness)
  lines(part_four_dates, part_four$Isol, type=lines_plot, lty=solid_
     line, col=gray, lwd = line_thickness)
  lines(part_four_dates, part_four$CI, type=lines_plot, lty=dashed_line
     , col=red, lwd = line_thickness)

  legend("topright", legend=column_names, col=column_colors, lty =
     column_types)
}
```

A.2 Basic SEIR model

Listing A.2: seir.R

```
require(deSolve)
```

```

# Basic SEIR model
SEIR <- function(time, current_state, params){
  with(as.list(c(current_state, params)), {
    N <- S + E + I + R
    dS <- -(beta * S * I) / N
    dE <- (beta * S * I) / N - sigma * E
    dI <- sigma * E - gamma * I
    dR <- gamma * I

    return(list(c(dS, dE, dI, dR)))
  })
}

# Setting some values to have an example plot
params <- c(
  sigma = 0.26,
  gamma = 0.15,
  beta = 0.5
)
initial_state <- c(S = 100, E = 0, I = 1, R = 0)
times <- 0:100

model <- ode(initial_state, times, SEIR, params)
df <- as.data.frame(model)
df$CI <- df$I + df$R

seir_plot(df,"An example of a SEIR model")
```

A.3 Transition equation correction

Listing A.3: beta.R

```
beta_func_original <- function(beta_0, beta_f, t, t_trans, k) {
  beta_0 - (beta_0 - beta_f) / (1 + exp((t - t_trans) / k))
}
beta_func <- function(beta_0, beta_f, t, t_trans, k) {
  beta_0 - (beta_0 - beta_f) / (1 + exp((t_trans - t) / k))
}

beta_0 <- 0.6
beta_f <- 0.6 * 0.11
t_trans <- 23
k <- 2

# Beta function with values fitted from the paper
```

```r
beta_parametrized_original <- function (t) {
  beta_func_original(beta_0 = beta_0, beta_f = beta_f, t = t, t_trans =
      t_trans, k = k)
}

beta_parametrized <- function (t) {
  beta_func(beta_0 = beta_0, beta_f = beta_f, t = t, t_trans = t_trans,
      k = k)
}

times <- 1:50
df <- data.frame(Original = rep(1, 50), Corrected = rep(1, 50))
df$Original <- sapply(times, beta_parametrized_original)
df$Corrected <- sapply(times, beta_parametrized)

beta_plot(df, "Transition equation correction")
```

Appendix B: Base model

B.1 Fitted parameters

Listing B.1: fitted_parameters.R

```r
# Result of the fitting
# Run before plotting any of the phases

params <- c(
  sigma_s = 0.22841296,
  sigma_a = 0.12872989,
  gamma_sp = 0.19536794,
  gamma_ap = 0.12500000,
  gamma_sn = 0.033460557,
  gamma_an = 0.03703704,
  omega = 0.70060133,
  omega_0 = 0.86611411,
  alpha = 0.29181595,
  beta_0 = 1,
  beta_f = 1 * 0.22213981,
  t_trans = 28.73244939,
  k = 0.01,
  rho = 1 / 14,
  sensitivity_pcr = 0.9,
  p = 0.36044370
)
```

B.2 Phase one

Listing B.2: phase_one.R

```r
require(deSolve)

# SEIR model implementation for phase 1
# Drawing an example plot for the test

phase_one <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E + I_s + I_a + R_pa + R_ps + R_n + H

    dS <- -beta * S * (I_s + I_a) / N
```

```
13
14      dE <- beta * S * (I_s + I_a) / N - sigma_s * E - sigma_a * E
15
16      dI_s <- (sigma_s * E - gamma_sp * I_s - omega_0 * I_s)
17      dI_a <- sigma_a * E - gamma_ap * I_a
18
19      dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
20      dR_pa <- gamma_ap * I_a - gamma_an * R_pa
21      dR_n <- gamma_sn * R_ps + gamma_an * R_pa
22
23      dH <- omega_0 * I_s
24
25      return(list(c(dS, dE, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH)))
26    })
27  }
28
29
30  phase_one_times <- 1:80
31
32  starting_state <- c(
33    S = 444,
34    E = 0,
35    I_s = 1,
36    I_a = 0,
37    R_ps = 0,
38    R_pa = 0,
39    R_n = 0,
40    H = 0
41  )
42
43  model <- ode(starting_state, phase_one_times, phase_one, params)
44
45  # Combining compartments into S-E-I-R
46  df <- as.data.frame(model)
47  df <- within(df, I <- I_s + I_a)
48  df <- within(df, R <- R_ps + R_pa + R_n)
49  df <- within(df, Isol <- H)
50  df <- within(df, CI <- I + R + Isol)
51  df <- subset(df, select = -c(I_s, I_a, R_ps, R_pa, R_n, H))
52
53  seir_plot(df, header = "Base model, Phase One (extended)", starting_
      date = as.Date("2020-03-14"))
```

B.3 Phase two

Listing B.3: phase_two.R

```r
require(deSolve)

# SEIR model implementation for phase 2
# Drawing an example plot for the test

phase_two <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E + I_s + I_a + R_pa + R_ps + R_n + Q + H

    dS <- -beta * S * (I_s + I_a) / N

    dE <- beta * S * (I_s + I_a) / N - sigma_s * E - sigma_a * E

    dI_s <- sigma_s * E - gamma_sp * I_s - alpha * I_s
    dI_a <- sigma_a * E - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa

    dQ <- -omega * Q + alpha * I_s
    dH <- omega * Q

    return(list(c(dS, dE, dI_s, dI_a, dR_ps, dR_pa, dR_n, dQ, dH)))
  })
}

starting_state <- c(
  S = 444,
  E = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0,
  Q = 0
)

phase_two_times <- 1:80
model <- ode(starting_state, phase_two_times, phase_two, params)

# Combining compartments into S-E-I-R
```

```
df <- as.data.frame(model)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n)
df <- within(df, Isol <- H)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(I_s, I_a, R_ps, R_pa, R_n, H, Q))

seir_plot(df, header = "Base model, Phase Two (extended)", starting_
    date = as.Date("2020-03-14"))
```

B.4 Phase three

Listing B.4: phase_three.R

```
require(deSolve)

# SEIR model implementation for phase 3
# Drawing an example plot for the test

phase_three <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E + I_s + I_a + R_pa + R_ps + R_n + H

    dS <- -beta * S * (I_s + I_a) / N

    dE <- beta * S * (I_s + I_a) / N - sigma_s * E - sigma_a * E

    dI_s <- sigma_s * E - gamma_sp * I_s - alpha * I_s
    dI_a <- sigma_a * E - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa

    dH <- alpha * I_s

    return(list(c(dS, dE, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH)))
  })
}

phase_three_times <- 1:80
starting_state <- c(
  S = 444,
  E = 0,
```

```
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0
)

model <- ode(starting_state, phase_three_times, phase_three, params)

# Combining compartments into S-E-I-R
df <- as.data.frame(model)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n)
df <- within(df, Isol <- H)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(I_s, I_a, R_ps, R_pa, R_n, H))

seir_plot(df, header = "Base model, Phase Three (extended)", starting_
    date = as.Date("2020-03-14"))
```

B.5 Phase four

Listing B.5: phase_four.R

```
require(deSolve)

# SEIR model implementation for phase 4
# Drawing an example plot for the test

phase_four <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E + I_s + I_a + R_pa + R_ps + R_n + H + Q

    dS <- -beta * S * (I_s + I_a)  / N

    dE <- beta * S * (I_s + I_a) / N - sigma_s * E - sigma_a * E

    dI_s <- sigma_s * E - gamma_sp * I_s - alpha * I_s
    dI_a <- sigma_a * E - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa + rho * Q
```

```
    dH <- alpha * I_s
    dQ <- -rho * Q

    return(list(c(dS, dE, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH, dQ)))
  })
}

phase_four_times <- 1:80
starting_state <- c(
  S = 444,
  E = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0,
  Q = 10
)

model <- ode(starting_state, phase_four_times, phase_four, params)

# Combining compartments into S-E-I-R
df <- as.data.frame(model)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n + H)
df <- within(df, Isol <- Q)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(I_s, I_a, R_ps, R_pa, R_n, H, Q))

seir_plot(df, header = "Base model, Phase Four (extended)", starting_
    date = as.Date("2020-03-14"))
```

B.6 Shelter model

Listing B.6: shelter.R

```
require(deSolve)

# Chicago shelter base model

shelter <- function(starting_state, params){
  state <- starting_state

  # Phase 1 from March 14 to March 29 (16 days)
```

```r
phase_one_times <- 1:16
model <- ode(state, phase_one_times, phase_one, params)
df <- as.data.frame(model)
# Saving last state for the next phase
last_state <- tail(df, n = 1)

# Combining compartments into SEIR
df <- within(df, Q <- 0)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n + H)
df <- within(df, Isol <- 0)

# Saving phase one results into a common table
result <- data.frame(df)

# Phase 2 from March 30 to April 4 (6 days)
phase_two_times <- 16:22
state <- c(
  S = last_state$S,
  E = last_state$E,
  I_s = last_state$I_s,
  I_a = last_state$I_a,
  R_ps = last_state$R_ps,
  R_pa = last_state$R_pa,
  R_n = last_state$R_n,
  Q = 0,
  H = last_state$H
)

model <- ode(state, phase_two_times, phase_two, params)
df <- as.data.frame(model)[-1, ]

# Saving last state for the next phase
last_state <- tail(df, n = 1)

# Combining compartments into SEIR
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n + Q + H)
df <- within(df, Isol <- 0)

result <- rbind(result, df)

# Phase 3 from April 5 to April 7 (3 days)
phase_three_times <- 22:25
state <- c(
  S = last_state$S,
  E = last_state$E,
```

```
    I_s = last_state$I_s,
    I_a = last_state$I_a,
    R_ps = last_state$R_ps,
    R_pa = last_state$R_pa,
    R_n = last_state$R_n,
    H = last_state$H + last_state$Q
  )

  model <- ode(state, phase_three_times, phase_three, params)
  df <- as.data.frame(model)[-1, ]

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, Q <- 0)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- 0)

  result <- rbind(result, df)

  # Phase 4.1 from April 8 to April 12 (5 days)
  phase_four_1_times <- 25:30
  state <- c(
    S = last_state$S,
    E = last_state$E,
    I_s = last_state$I_s,
    I_a = last_state$I_a,
    R_ps = last_state$R_ps,
    R_pa = last_state$R_pa,
    R_n = last_state$R_n,
    H = last_state$H,
    Q = 0
  )

  model <- ode(state, phase_four_1_times, phase_four, params)
  df <- as.data.frame(model)[-1, ]

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

```

```
  result <- rbind(result, df)

  # Phase 4.2 from April 13 to April 18 (7 days)
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR * n_
      individuals in each compartment on test day)
  # to Q (Isol)
  sensitivity_pcr <- unname(params["sensitivity_pcr"])
  phase_four_2_times <- 30:36
  state <- c(
    S = last_state$S,
    E = last_state$E,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I
        _a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_2_times, phase_four, params)
  df <- as.data.frame(model)

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)

  # Phase 4.3 from April 18 to April 30 (13 days)
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR *
      nindividuals in each compartment on test day)
  # to Q (Isol)
  phase_four_3_times <- 36:48
  state <- c(
    S = last_state$S,
    E = last_state$E,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
```

```
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I
        _a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_3_times, phase_four, params)
  df <- as.data.frame(model)

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)

  # Phase 4.4 from April 30 to May 13 (14 days)
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR *
      nindividuals in each compartment on test day)
  # to Q (Isol)
  phase_four_4_times <- 48:61
  state <- c(
    S = last_state$S,
    E = last_state$E,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I
        _a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_4_times, phase_four, params)
  df <- as.data.frame(model)

  # Combining compartments into SEIR
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)
  rownames(result) <- NULL
  result <- within(result, CI <- I + R + Isol)
```

```
  return(result)
}

starting_state <- c(
  S = 444,
  E = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0
)

result <- shelter(starting_state, params)
result <- subset(result, select = -c(I_s, I_a, R_ps, R_pa, R_n, H, Q))
shelter_plot(result, header = "Chicago Shelter COVID-19 Outbreak, Base
    Model")
```

B.7 Parameters fitting

Listing B.7: fit.R

```
require(deSolve)

# Error function to be minimized
rmsle <- function(x) {
  current_params <- c(
    beta_0 = x[1],
    beta_f = x[1] * x[2],
    omega_0 = x[3],
    t_trans = x[4],
    k = x[5],
    p = x[6],

    sigma_s = 1 / x[7] * (1 - x[6]),
    sigma_a = 1 / x[7] * x[6],

    gamma_sp = 1 / x[8],
    gamma_ap = 1 / x[9],

    gamma_sn = 1 / (x[10] - x[8]),
    gamma_an = 1 / (x[11] - x[9]),
    alpha = x[12],
    omega = x[13],
    sensitivity_pcr = x[14],
```

```
    rho = 1 / 14
  )

  state <- c(
    S = 444,
    E = 0,
    I_s = 1,
    I_a = 0,
    R_ps = 0,
    R_pa = 0,
    R_n = 0,
    H = 0
  )

  df <- shelter(state, current_params)
  sensitivity_pcr <- unname(current_params["sensitivity_pcr"])

  result <- c(
      (log1p(df$CI[1]) - log1p(1))^2,
      (log1p(df$CI[2]) - log1p(1))^2,
      (log1p(df$CI[3]) - log1p(1))^2,
      (log1p(df$CI[4]) - log1p(2))^2,
      (log1p(df$CI[5]) - log1p(3))^2,
      (log1p(df$CI[6]) - log1p(3))^2,
      (log1p(df$CI[7]) - log1p(4))^2,
      (log1p(df$CI[8]) - log1p(4))^2,
      (log1p(df$CI[9]) - log1p(5))^2,
      (log1p(df$CI[10]) - log1p(6))^2,
      (log1p(df$CI[11]) - log1p(6))^2,
      (log1p(df$CI[12]) - log1p(6))^2,
      (log1p(df$CI[13]) - log1p(7))^2,
      (log1p(df$CI[14]) - log1p(9))^2,
      (log1p(df$CI[15]) - log1p(9))^2,
      (log1p(df$CI[16]) - log1p(9))^2,
      (log1p(df$CI[17]) - log1p(10))^2
  )
  result <- c(result, (log1p(df$CI[22] - df$CI[17]) - log1p(18)) ^ 2)
  result <- c(result, (log1p(df$H[22] - df$H[17]) - log1p(7)) ^ 2)

  result <- c(result, (log1p(df$H[25] - df$H[22]) - log1p(26)) ^ 2)

  result <- c(result,
      2 * (log1p(sensitivity_pcr * (df$I_s[30] + df$I_a[30] + df$R_ps
          [30] + df$R_pa[30])) - log1p(166)) ^ 2,
      2 * (log1p(sensitivity_pcr * (df$I_s[37] + df$I_a[37] + df$R_ps
          [37] + df$R_pa[37])) - log1p(24)) ^ 2,
```

```
      2 * (log1p(sensitivity_pcr * (df$I_s[50] + df$I_a[50] + df$R_ps
          [50] + df$R_pa[50])) - log1p(23)) ^ 2,
      2 * (log1p(sensitivity_pcr * (df$I_s[61] + df$I_a[61] + df$R_ps
          [61] + df$R_pa[61])) - log1p(1)) ^ 2
  )
  result <- c(result,
      (log1p(df$H[37] - df$H[31]) - log1p(20)) ^ 2,
      (log1p(df$H[50] - df$H[38]) - log1p(4)) ^ 2,
      (log1p(df$H[61] - df$H[51]) - log1p(0)) ^ 2
  )
  return(sqrt(abs(mean(result))))
}
# Fitting following params:

# beta_0 - initial Beta
# beta_f_pct - final Beta as a percentage of beta_0
# omega_0 - Rate of hospital admission of Infectious symptomatic
    persons before screening
# t_trans - Day where beta reaches halfway between beta_0 and beta_f
# k - Rate of transformation of beta
# p - Asymptomatic percentage

# incubation_period - Time between E and I compartments (days)
# sigma_s = 1/(incubation period) * (% symptomatic),
# sigma_a = 1/(incubation period) * (% asymptomatic)

# Infectious period for symptomatic persons (days)
# gamma_sp = 1/(infectious period for symptomatic persons)

# Infectious period for asymptomatic persons (days)
# gamma_ap = 1/(infectious period for asymptomatic persons)

# Period of RT- PCR-positivity for symptomatic persons (days)
# gamma_sn = 1/[(duration of RT-PCR-positivity for symptomatic persons)
     - (infectious period)]

# Period of RT- PCR-positivity for asymptomatic persons (days)
# gamma_an = 1/[(duration of RT-PCR-positivity for asymptomatic persons
    ) - (infectious period)]

# alpha - Rate of detection of symptomatic infectious persons through
    screening
# omega - Rate of hospital admission of Isolsoft symptomatic persons
    during phase 2
# sensitivity_pcr

# starting values are mean of the boundaries
```

```r
lower_params <- c(0, 0, 0.05, 1, 0.01, 0.18, 2.8, 3, 3, 16, 3, 0.01,
    0.05, 0.72)
upper_params <- c(1, 1, 1.0, 50, 2, 0.87, 4.0, 8, 8, 35, 35, 1, 1, 0.9)
starting_params <- (lower_params + upper_params) / 2

result <- optim(
  par = starting_params,
  fn = rmsle,
  gr = NULL,
  control = list(maxit = 100000, pgtol = 1e-16, retol=1e-16),
  method = "L-BFGS-B",
  lower = lower_params,
  upper = upper_params,
  hessian = TRUE
)

result_params <- c(
  beta_0 = result$par[1],
  beta_f_pct = result$par[2],
  omega_0 = result$par[3],
  t_trans = result$par[4],
  k = result$par[5],
  p = result$par[6],

  sigma_s = 1 / result$par[7] * (1 - result$par[6]),
  sigma_a = 1 / result$par[7] * result$par[6],

  gamma_sp = 1 / result$par[8],
  gamma_ap = 1 / result$par[9],

  gamma_sn = 1 / (result$par[10] - result$par[8]),
  gamma_an = 1 / (result$par[11] - result$par[9]),
  alpha = result$par[12],
  omega = result$par[13],
  sensitivity_pcr = result$par[14]
)

print(result)
print(result_params)
```

Appendix C: Alternative model

C.1 Fitted parameters

Listing C.1: fitted_parameters.R

```
params <- c(
  sigma_s = 0.35714286,
  sigma_a = 0.25000000,
  gamma_sp = 0.12500000,
  gamma_ap = 0.12500000,
  gamma_sn = 0.03703704,
  gamma_an = 0.03703704,
  omega = 0.56447563,
  omega_0 = 0.83792416,
  alpha = 0.37555100,
  beta_0 = 1,
  beta_f = 1 * 0.07572578,
  t_trans = 29.15235226,
  k = 0.01,
  rho = 1 / 14,
  sensitivity_pcr = 0.9,
  p = 0.44804812
)
```

C.2 Phase one

Listing C.2: phase_one.R

```
require(deSolve)

# Alternative SEIR model implementation for phase 1
# Drawing an example plot for the test

phase_one_alternative <- function(time, current_state, params){

  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E_s + E_a + I_s + I_a + R_pa + R_ps + R_n + H

    dS <- -beta * S * (I_s + I_a) / N

    dE_s <- (1 - p) * beta * S * (I_s + I_a) / N - sigma_s * E_s
```

```
    dE_a <- p * beta * S * (I_s + I_a) / N - sigma_a * E_a

    dI_s <- sigma_s * E_s - gamma_sp * I_s - omega_0 * I_s
    dI_a <- sigma_a * E_a - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa

    dH <- omega_0 * I_s

    return(list(c(dS, dE_s, dE_a, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH)))
  })
}

phase_one_times <- 0:80
starting_state <- c(
  S = 444,
  E_s = 0,
  E_a = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0
)

model <- ode(starting_state, phase_one_times, phase_one_alternative,
    params)

# Combining compartments into SEIR
df <- as.data.frame(model)
df <- within(df, I <- I_s + I_a)
df <- within(df, E <- E_s + E_a)
df <- within(df, R <- R_ps + R_pa + R_n)
df <- within(df, Isol <- H)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(E_s, E_a, I_s, I_a, R_ps, R_pa, R_n, H))

seir_plot(df, "Alternative Model, Phase One (Extended)", starting_date
    = as.Date("2020-03-14"))
```

C.3 Phase two

Listing C.3: phase_two.R

```r
require(deSolve)

# Alternative SEIR model implementation for phase 2
# Drawing an example plot for the test

phase_two_alternative <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E_s + E_a + I_s + I_a + R_pa + R_ps + R_n + Q + H

    dS <- -beta * S * (I_s + I_a) / N

    dE_s <- (1 - p) * beta * S * (I_s + I_a) / N - sigma_s * E_s
    dE_a <- p * beta * S * (I_s + I_a) / N - sigma_a * E_a

    dI_s <- sigma_s * E_s - gamma_sp * I_s - alpha * I_s
    dI_a <- sigma_a * E_a - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa

    dQ <- -omega * Q + alpha * I_s
    dH <- omega * Q

    return(list(c(dS, dE_s, dE_a, dI_s, dI_a, dR_ps, dR_pa, dR_n, dQ,
        dH)))
  })
}

phase_two_times <- 1:80
starting_state <- c(
  S = 444,
  E_s = 0,
  E_a = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0,
  Q = 0
)

```

```
model <- ode(starting_state, phase_two_times, phase_two_alternative,
    params)

# Combining compartments into S-E-I-R
df <- as.data.frame(model)
df <- within(df, E <- E_s + E_a)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n)
df <- within(df, Isol <- H)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(E_s, E_a, I_s, I_a, R_ps, R_pa, R_n, H, Q)
    )

seir_plot(df, header = "Alternative Model, Phase Two (Extended)",
    starting_date = as.Date("2020-03-14"))
```

C.4 Phase three

Listing C.4: phase_three.R

```
require(deSolve)

# Alternative SEIR model implementation for phase 3
# Drawing an example plot for the test

phase_three_alternative <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E_s + E_a + I_s + I_a + R_pa + R_ps + R_n + H

    dS <- -beta * S * (I_s + I_a) / N

    dE_s <- (1 - p) * beta * S * (I_s + I_a) / N - sigma_s * E_s
    dE_a <- p * beta * S * (I_s + I_a) / N - sigma_a * E_a

    dI_s <- sigma_s * E_s - gamma_sp * I_s - alpha * I_s
    dI_a <- sigma_a * E_a - gamma_ap * I_a

    dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
    dR_pa <- gamma_ap * I_a - gamma_an * R_pa
    dR_n <- gamma_sn * R_ps + gamma_an * R_pa

    dH <- alpha * I_s

    return(list(c(dS, dE_s, dE_a, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH)))
```

```
  })
}

phase_three_times <- 1:80
state <- c(
  S = 444,
  E_s = 0,
  E_a = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0
)

model <- ode(state, phase_three_times, phase_three_alternative, params)

# Combining compartments into S-E-I-R
df <- as.data.frame(model)
df <- within(df, E <- E_s + E_a)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n)
df <- within(df, Isol <- H)
df <- within(df, CI <- I + R + Isol)
df <- subset(df, select = -c(E_a, E_s, I_s, I_a, R_ps, R_pa, R_n, H))

seir_plot(df, header = "Alternative Model, Phase Three (Extended)",
    starting_date = as.Date("2020-03-14"))
```

C.5 Phase four

Listing C.5: phase_four.R

```
require(deSolve)

# Alternative SEIR model implementation for phase 4
# Drawing an example plot for the test

phase_four_alternative <- function(time, current_state, params){
  with(as.list(c(current_state, params)),{
    beta <- beta_func(beta_0, beta_f, time, t_trans, k)

    N <- S + E_s + E_a + I_s + I_a + R_pa + R_ps + R_n + H + Q

    dS <- -beta * S * (I_s + I_a)  / N
```

```
13
14      dE_s <- (1 - p) * beta * S * (I_s + I_a) / N - sigma_s * E_s
15      dE_a <- p * beta * S * (I_s + I_a) / N - sigma_a * E_a
16
17      dI_s <- sigma_s * E_s - gamma_sp * I_s - alpha * I_s
18      dI_a <- sigma_a * E_a - gamma_ap * I_a
19
20      dR_ps <- gamma_sp * I_s - gamma_sn * R_ps
21      dR_pa <- gamma_ap * I_a - gamma_an * R_pa
22      dR_n <- gamma_sn * R_ps + gamma_an * R_pa + rho * Q
23
24      dH <- alpha * I_s
25      dQ <- -rho * Q
26
27      return(list(c(dS, dE_s, dE_a, dI_s, dI_a, dR_ps, dR_pa, dR_n, dH,
            dQ)))
28    })
29  }
30
31  phase_four_times <- 1:80
32  state <- c(
33    S = 444,
34    E_s = 0,
35    E_a = 0,
36    I_s = 1,
37    I_a = 0,
38    R_ps = 0,
39    R_pa = 0,
40    R_n = 0,
41    H = 0,
42    Q = 10
43  )
44
45  model <- ode(state, phase_four_times, phase_four_alternative, params)
46
47  # Combining compartments into S-E-I-R
48  df <- as.data.frame(model)
49  df <- within(df, E <- E_s + E_a)
50  df <- within(df, I <- I_s + I_a)
51  df <- within(df, R <- R_ps + R_pa + R_n + H)
52  df <- within(df, Isol <- Q)
53  df <- within(df, CI <- I + R + Isol)
54  df <- subset(df, select = -c(E_s, E_a, I_s, I_a, R_ps, R_pa, R_n, H, Q)
        )
55
56  seir_plot(df, header = "Alternative Model, Phase Four (Extended)",
        starting_date = as.Date("2020-03-14"))
```

C.6 Shelter model

Listing C.6: shelter.R

```r
require(deSolve)
shelter_alternative <- function(starting_state, params){
  state <- starting_state

  # Phase 1 from March 14 to March 29 (16 days)
  phase_one_times <- 1:16
  model <- ode(state, phase_one_times, phase_one_alternative, params)
  df <- as.data.frame(model)
  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, Q <- 0)
  df <- within(df, E <- E_s + E_a)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- 0)

  # Saving phase one results into a common table
  result <- data.frame(df)

  # Phase 2 from March 30 to April 4 (6 days)
  phase_two_times <- 16:22
  state <- c(
    S = last_state$S,
    E_s = last_state$E_s,
    E_a = last_state$E_a,
    I_s = last_state$I_s,
    I_a = last_state$I_a,
    R_ps = last_state$R_ps,
    R_pa = last_state$R_pa,
    R_n = last_state$R_n,
    Q = 0,
    H = last_state$H
  )

  model <- ode(state, phase_two_times, phase_two_alternative, params)
  df <- as.data.frame(model)[-1, ]

  # Saving last state for the next phase
```

```r
last_state <- tail(df, n = 1)

# Combining compartments into SEIR
df <- within(df, E <- E_s + E_a)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n + Q + H)
df <- within(df, Isol <- 0)

result <- rbind(result, df)

# Phase 3 from April 5 to April 7 (3 days)
phase_three_times <- 22:25
state <- c(
  S = last_state$S,
  E_s = last_state$E_s,
  E_a = last_state$E_a,
  I_s = last_state$I_s,
  I_a = last_state$I_a,
  R_ps = last_state$R_ps,
  R_pa = last_state$R_pa,
  R_n = last_state$R_n,
  H = last_state$H + last_state$Q
)

model <- ode(state, phase_three_times, phase_three_alternative,
    params)
df <- as.data.frame(model)[-1, ]

# Saving last state for the next phase
last_state <- tail(df, n = 1)

# Combining compartments into SEIR
df <- within(df, Q <- 0)
df <- within(df, E <- E_s + E_a)
df <- within(df, I <- I_s + I_a)
df <- within(df, R <- R_ps + R_pa + R_n + H)
df <- within(df, Isol <- 0)

result <- rbind(result, df)

# Phase 4.1 from April 8 to April 12 (5 days)
phase_four_1_times <- 25:30
state <- c(
  S = last_state$S,
  E_s = last_state$E_s,
  E_a = last_state$E_a,
  I_s = last_state$I_s,
```

```
    I_a = last_state$I_a,
    R_ps = last_state$R_ps,
    R_pa = last_state$R_pa,
    R_n = last_state$R_n,
    H = last_state$H,
    Q = 0
  )

  model <- ode(state, phase_four_1_times, phase_four_alternative,
     params)
  df <- as.data.frame(model)[-1, ]

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, E <- E_s + E_a)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)

  # Phase 4.2 from April 13 to April 18 (7 days)
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR *
     nindividuals in each compartment on test day)
  # to Q (Isol)
  sensitivity_pcr <- unname(params["sensitivity_pcr"])
  phase_four_2_times <- 30:36
  state <- c(
    S = last_state$S,
    E_s = last_state$E_s,
    E_a = last_state$E_a,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I
       _a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_2_times, phase_four_alternative,
     params)
  df <- as.data.frame(model)

```

```r
  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, E <- E_s + E_a)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)

  # Phase 4.3 from April 18 to April 30 (13 days)
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR * nindividuals in each compartment on test day)
  # to Q (Isol)
  phase_four_3_times <- 36:48
  state <- c(
    S = last_state$S,
    E_s = last_state$E_s,
    E_a = last_state$E_a,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I_a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_3_times, phase_four_alternative, params)
  df <- as.data.frame(model)

  # Saving last state for the next phase
  last_state <- tail(df, n = 1)

  # Combining compartments into SEIR
  df <- within(df, E <- E_s + E_a)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)

  # Phase 4.4 from April 30 to May 13 (14 days)
```

```
  # I_s, I_a, R_ps, R_pa comparments lose (SensitivityPCR * nindividuals in each compartment on test day)
  # to Q (Isol)
  phase_four_4_times <- 48:61
  state <- c(
    S = last_state$S,
    E_s = last_state$E_s,
    E_a = last_state$E_a,
    I_s = last_state$I_s * (1 - sensitivity_pcr),
    I_a = last_state$I_a * (1 - sensitivity_pcr),
    R_ps = last_state$R_ps * (1 - sensitivity_pcr),
    R_pa = last_state$R_pa * (1 - sensitivity_pcr),
    R_n = last_state$R_n,
    H = last_state$H,
    Q = last_state$Q + sensitivity_pcr * (last_state$I_s + last_state$I_a + last_state$R_ps + last_state$R_pa)
  )

  model <- ode(state, phase_four_4_times, phase_four_alternative, params)
  df <- as.data.frame(model)

  # Combining compartments into SEIR
  df <- within(df, E <- E_s + E_a)
  df <- within(df, I <- I_s + I_a)
  df <- within(df, R <- R_ps + R_pa + R_n + H)
  df <- within(df, Isol <- Q)

  result <- rbind(result, df)
  rownames(result) <- NULL
  result <- within(result, CI <- I + R + Isol)
  return(result)
}

starting_state <- c(
  S = 444,
  E_s = 0,
  E_a = 0,
  I_s = 1,
  I_a = 0,
  R_ps = 0,
  R_pa = 0,
  R_n = 0,
  H = 0
)

result <- shelter_alternative(starting_state, params)
```

```
result <- subset(result, select = -c(E_a, E_s, I_s, I_a, R_ps, R_pa, R_
    n, H, Q))
shelter_plot(result, header = "Chicago Shelter COVID-19 Outbreak,
    Alternative Model")
```

C.7 Parameters fitting

Listing C.7: fit.R

```
require(deSolve)

rmsle_imrpoved <- function(x) {
  params <- c(
    beta_0 = x[1],
    beta_f = x[1] * x[2],
    omega_0 = x[3],
    t_trans = x[4],
    k = x[5],
    p = x[6],

    sigma_s = 1 / x[7],
    sigma_a = 1 / x[8],

    gamma_sp = 1 / x[9],
    gamma_ap = 1 / x[10],

    gamma_sn = 1 / (x[11] - x[9]),
    gamma_an = 1 / (x[12] - x[10]),
    alpha = x[13],
    omega = x[14],
    sensitivity_pcr = x[15],
    rho = 1 / 14
  )

  state <- c(
    S = 444,
    E_s = 0,
    E_a = 0,
    I_s = 1,
    I_a = 0,
    R_ps = 0,
    R_pa = 0,
    R_n = 0,
    H = 0
  )
  df <- shelter_alternative(state, params)
```

```
  sensitivity_pcr <- unname(params["sensitivity_pcr"])

  result <- c(
    (log1p(df$CI[1]) - log1p(1))^2,
    (log1p(df$CI[2]) - log1p(1))^2,
    (log1p(df$CI[3]) - log1p(1))^2,
    (log1p(df$CI[4]) - log1p(2))^2,
    (log1p(df$CI[5]) - log1p(3))^2,
    (log1p(df$CI[6]) - log1p(3))^2,
    (log1p(df$CI[7]) - log1p(4))^2,
    (log1p(df$CI[8]) - log1p(4))^2,
    (log1p(df$CI[9]) - log1p(5))^2,
    (log1p(df$CI[10]) - log1p(6))^2,
    (log1p(df$CI[11]) - log1p(6))^2,
    (log1p(df$CI[12]) - log1p(6))^2,
    (log1p(df$CI[13]) - log1p(7))^2,
    (log1p(df$CI[14]) - log1p(9))^2,
    (log1p(df$CI[15]) - log1p(9))^2,
    (log1p(df$CI[16]) - log1p(9))^2,
    (log1p(df$CI[17]) - log1p(10))^2
  )
  result <- c(result, (log1p(df$CI[22] - df$CI[17]) - log1p(18)) ^ 2)
  result <- c(result, (log1p(df$H[22] - df$H[17]) - log1p(7)) ^ 2)

  result <- c(result, (log1p(df$H[25] - df$H[22]) - log1p(26)) ^ 2)

  result <- c(result,
              2 * (log1p(sensitivity_pcr * (df$I_s[30] + df$I_a[30] +
                 df$R_ps[30] + df$R_pa[30])) - log1p(166)) ^ 2,
              2 * (log1p(sensitivity_pcr * (df$I_s[37] + df$I_a[37] +
                 df$R_ps[37] + df$R_pa[37])) - log1p(24)) ^ 2,
              2 * (log1p(sensitivity_pcr * (df$I_s[50] + df$I_a[50] +
                 df$R_ps[50] + df$R_pa[50])) - log1p(23)) ^ 2,
              2 * (log1p(sensitivity_pcr * (df$I_s[61] + df$I_a[61] +
                 df$R_ps[61] + df$R_pa[61])) - log1p(1)) ^ 2
  )
  result <- c(result,
              (log1p(df$H[37] - df$H[31]) - log1p(20)) ^ 2,
              (log1p(df$H[50] - df$H[38]) - log1p(4)) ^ 2,
              (log1p(df$H[61] - df$H[51]) - log1p(0)) ^ 2
  )
  return(sqrt(abs(mean(result))))

}
# Fitting following params:

# beta_0 - initial Beta
```

```
# beta_f_pct - final Beta as a percentage of beta_0
# omega_0 - Rate of hospital admission of Infectious symptomatic persons before screening
# t_trans - Day where beta reaches halfway between beta_0 and beta_f
# k - Rate of transformation of beta
# p - Asymptomatic percentage

# incubation_period (symptomatic) - Time between E_s and I_s compartments (days)
# sigma_s = 1/(incubation period) * (% symptomatic),

# incubation_period (asymptomatic) - Time between E_a and I_a compartments (days)
# sigma_a = 1/(incubation period) * (% asymptomatic)

# Infectious period for symptomatic persons (days)
# gamma_sp = 1/(infectious period for symptomatic persons)

# Infectious period for asymptomatic persons (days)
# gamma_ap = 1/(infectious period for asymptomatic persons)

# Period of RT- PCR-positivity for symptomatic persons (days)
# gamma_sn = 1/[(duration of RT-PCR-positivity for symptomatic persons) - (infectious period)]

# Period of RT- PCR-positivity for asymptomatic persons (days)
# gamma_an = 1/[(duration of RT-PCR-positivity for asymptomatic persons) - (infectious period)]

# alpha - Rate of detection of symptomatic infectious persons through screening
# omega - Rate of hospital admission of Isolsoft symptomatic persons during phase 2
# sensitivity_pcr

# starting values are mean of the boundaries
lower_params <- c(0, 0, 0.05, 1, 0.01, 0.18, 2.8, 2.8, 3, 3, 16, 3, 0.01, 0.05, 0.72)
upper_params <- c(1, 1, 1.0, 50, 2, 0.87, 4.0, 4.0, 8, 8, 35, 35, 1, 1, 0.9)
starting_params <- (lower_params + upper_params) / 2

result <- optim(
  par = starting_params,
  fn = rmsle_imrpoved,
  gr = NULL,
  control = list(maxit = 100000),
```

```
  method = "L-BFGS-B",
  lower = lower_params,
  upper = upper_params,
  hessian = TRUE
)

result_params <- c(
  beta_0 = result$par[1],
  beta_f_pct = result$par[2],
  omega_0 = result$par[3],
  t_trans = result$par[4],
  k = result$par[5],
  p = result$par[6],

  sigma_s = 1 / result$par[7],
  sigma_a = 1 / result$par[8],

  gamma_sp = 1 / result$par[9],
  gamma_ap = 1 / result$par[10],

  gamma_sn = 1 / (result$par[11] - result$par[9]),
  gamma_an = 1 / (result$par[12] - result$par[10]),
  alpha = result$par[13],
  omega = result$par[14],
  sensitivity_pcr = result$par[15]
)
print(result)
print(result_params)
```

Bibliography

[1] Xiao AT, Tong YX, and Zhang S. “Profile of RT-PCR for SARS-CoV-2: A Preliminary Study From 56 COVID-19 Patients.” In: *Clin Infect Dis* (2020). DOI: doi:10.1093/cid/ciaa460. URL: https://pubmed.ncbi.nlm.nih.gov/32306036/.

[2] Tom Britton. “Stochastic epidemic models: A survey”. In: *Mathematical Biosciences* 225.1 (2010), pp. 24–35. ISSN: 0025-5564. DOI: https://doi.org/10.1016/j.mbs.2010.01.006. URL: https://www.sciencedirect.com/science/article/pii/S0025556410000143.

[3] Muge Cevik et al. “Virology, transmission, and pathogenesis of SARS-CoV-2”. In: *BMJ* 371 (2020). DOI: 10.1136/bmj.m3862. eprint: https://www.bmj.com/content/371/bmj.m3862.full.pdf. URL: https://www.bmj.com/content/371/bmj.m3862.

[4] Cevik, Muge, and et al. “SARS-CoV-2, SARS-CoV, and MERS-CoV viral load dynamics, duration of viral shedding, and infectiousness: a systematic review and meta-analysis.” In: *The Lancet. Microbe vol. 2,1* (2021). DOI: doi:10.1016/S2666-5247(20)30172-5. URL: https://pubmed.ncbi.nlm.nih.gov/33521734/.

[5] Yi-Shin Chang et al. “Transmission Dynamics of Large Coronavirus Disease Outbreak in Homeless Shelter, Chicago, Illinois, USA, 2020”. In: *Emerging Infectious Diseases* (2022).

[6] Centers for Disease Control and Prevention. *COVID-19 Forecasting and Mathematical Modeling*. 2023. URL: https://www.cdc.gov/coronavirus/2019-ncov/science/forecasting/forecasting-math-modeling.html.

[7] Centers for Disease Control and Prevention. *Factors That Affect Your Risk of Getting Very Sick from COVID-19*. 2023. URL: https://www.cdc.gov/coronavirus/2019-ncov/your-health/risks-getting-very-sick.html.

[8] Centers for Disease Control and Prevention. *Symptoms of COVID-19*. 2022. URL: https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/symptoms.html.

[9] Mosites E et al. “Assessment of SARS-CoV-2 Infection Prevalence in Homeless Shelters — Four U.S. Cities, March 27–April 15, 2020.” In: *MMWR Morb Mortal Wkly Rep* (2020). DOI: 10.15585/mmwr.mm6917e1. URL: https://www.cdc.gov/mmwr/volumes/69/wr/mm6917e1.htm.

[10] Tobolowsky FA et al. “COVID-19 Outbreak Among Three Affiliated Homeless Service Sites”. In: *MMWR Morb Mortal Wkly Rep* (2020). DOI: 10.15585/mmwr.mm6917e2external. URL: https://www.cdc.gov/mmwr/volumes/69/wr/mm6917e2.htm.

[11] Bill & Melinda Gates Foundation. *SEIR and SEIRS models*. 2023. URL: https://docs.idmod.org/projects/emod-hiv/en/2.20_a/model-seir.html.

[12] Ghinai I et al. “Risk Factors for Severe Acute Respiratory Syndrome Coronavirus 2 Infection in Homeless Shelters in Chicago, Illinois-March-May, 2020.” In: *Open Forum Infect Dis. 2020;7(11):ofaa477.* (2020). DOI: 10.1093/ofid/ofaa477. URL: 10.1093/ofid/ofaa477.

[13] Dong C. Liu and Jorge Nocedal. “On the limited memory BFGS method for large scale optimization”. In: *Mathematical Programming volume 45, pages 503–528 (1989)* (1989). DOI: https://doi.org/10.1007/BF01589116. URL: https://link.springer.com/article/10.1007/BF01589116#citeas.

[14] Yang Liu et al. “Viral dynamics in mild and severe cases of COVID-19”. In: *The Lancet. Infectious Diseases* (2020). DOI: doi:10.1016/S1473-3099(20)30232-2. URL: https://doi.org/10.1016/S1473-3099(20)30232-2.

[15] Qiuyue Ma et al. “Global Percentage of Asymptomatic SARS-CoV-2 Infections Among the Tested Population and Individuals With Confirmed COVID-19 Diagnosis: A Systematic Review and Meta-analysis”. In: *JAMA Network Open* (2021). DOI: 10.1001/jamanetworkopen.2021.37257. URL: https://doi.org/10.1001/jamanetworkopen.2021.37257.

[16] Garner MG and Hamilton SA. “Principles of epidemiological modelling”. In: *Revue scientifique et technique* (2011). DOI: doi:10.20506/rst.30.2.2045. URL: https://doi.org/10.20506/rst.30.2.2045.

[17] World Health Organization. *Coronavirus disease (COVID-19) pandemic*. 2023. URL: https://www.who.int/europe/emergencies/situations/covid-19.

[18] World Health Organization. *Coronavirus disease (COVID-19): How is it transmitted?* 2021. URL: https://www.who.int/news-room/questions-and-answers/item/coronavirus-disease-covid-19-how-is-it-transmitted.

[19] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2022. URL: https://www.R-project.org/.

[20] Sanche S et al. “High Contagiousness and Rapid Spread of Severe Acute Respiratory Syndrome Coronavirus 2”. In: *Emerging Infectious Diseases* (2020). DOI: doi:10.3201/eid2607.200282. URL: https://wwwnc.cdc.gov/eid/article/26/7/20-0282_article.

[21] California Institute of Technology. *How Does Disease Modeling Work?* 2023. URL: https://scienceexchange.caltech.edu/topics/covid-19-coronavirus-sars-cov-2/disease-modeling.

[22] Worldometer. *COVID-19 Coronavirus Pandemic*. 2023. URL: https://www.worldometers.info/coronavirus/.

Statutory Declaration in Lieu of an Oath

I – Dmitrii Rudanov – do hereby declare in lieu of an oath that I have composed the presented work independently on my own and without any other resources than the ones given.

All thoughts taken directly or indirectly from external sources are correctly acknowledged.

This work has neither been previously submitted to another authority nor has it been published yet.

Mittweida, 07. November 2023

Location, Date

Dmitrii Rudanov, B.Sc.