

The COVID-19 outbreak in Germany – Models and Parameter Estimation

Peter Heidrich^{1*}, Moritz Schäfer¹, Mostafa Nikouei¹, Thomas Götz¹

¹Mathematical Institute, University of Koblenz–Landau, 56070 Koblenz, Germany

*Email: heidrich@uni-koblenz.de

Abstract

Since the end of 2019 an outbreak of a new strain of coronavirus, called SARS–CoV–2, is reported from China and later also from other parts of the world. Since 21 January 2020, World Health Organization (WHO) reports daily data on confirmed cases and deaths from both China and other countries [1]. The Johns Hopkins University [2] collects those data from various sources worldwide on a daily basis. For Germany, the Robert–Koch–Institute (RKI) also issues daily reports on the current number of infections and infection related fatal cases and also provides estimates of several disease-related parameters [3]. In this work we present an extended SEIRD–model to describe these disease dynamics in Germany. The model takes into account the susceptible, exposed, infected, recovered and deceased fractions of the population. Epidemiological parameters like the transmission rate, lethality or the detection rate of infected individuals are estimated by fitting the model output to available data. For the parameter estimation itself we compare two methods: an adjoint based approach and a Monte–Carlo based Metropolis algorithm.

Keywords: COVID–19, Epidemiology, Disease dynamics, SEIRD–model, Parameter estimation, Adjoint equations, Metropolis algorithm.

2010 MSC classification number: 92D30.

1. INTRODUCTION

In December 2019, first cases of a pneumonia of unknown cause were reported from Wuhan, China. In the meantime, these cases were identified as infections with a novel strain of coronavirus, called SARS–CoV–2, and the disease it causes was called Coronavirus Disease 2019 (COVID–19). At the beginning of January 2020, the virus spread over mainland China and reached other provinces. From 21 January onwards, WHO's daily situation reports [1] or Johns Hopkins University [2] (JHU) contain the latest figures on confirmed cases and deaths for almost all countries. In this work we rely on the data published by the JHU due to their rapid updates and easy accessibility.

The first COVID–19 case in Germany was reported on 27 January 2020 in Bavaria. Later cases were imported by travelers from China, Iran or Italy as well as tourists returning from ski holidays in Austria and Italy. By 1 March 2020, more than 100 cases were reported in Germany; since then, the number of cases began to rise exponentially. The first deaths were reported on 9 March [3]. By 16 March, the federal government introduced first measures to reduce the spread of the disease: schools, kindergartens and universities were closed. On 22 March, these measures were tightened by implementing a national curfew and contact ban. People are advised to stay at home, leaving only for work related activities, necessary shopping, medical treatment or sports [4]. By mid of April, these mitigation measures showed some success with the number of new infections declining from its peak of 6,294 on 28 March to less than 1,000 from 2 May onwards. On 6 May, a relaxation of the imposed restrictions to social and economic life was announced. Since then, the federal states are progressing at an individual pace to "normality".

Asking the population to remain cautious and not to cause a second wave, local governments of cities or districts are in charge to reinforce restrictions in case the number of new infections surpasses the limit of 50 per 100,000 inhabitants within 7 days as of 6 May [5], [6]. Already four days later five districts exceeded this limit; with no measures reported to alleviate it.

*Corresponding author

Received June 26th, 2020, Revised June 28th, 2020, Accepted for publication June 30th, 2020. Copyright ©2020 Published by Indonesian Biomathematical Society, e-ISSN: 2549-2896, DOI:10.5614/cbms.2020.3.1.5

The pandemic continues to spread worldwide (as of June 2020) and the actual possibility of a second wave demands for models to predict epidemic scenarios for the near and mid future. The quality of those models heavily relies on the parameters used. In this study we present SEIRD-models which are some sort of quasi standard in epidemiological simulations and estimate their parameters by using the available data from the JHU. The estimation itself is based on a least-squares fit between the model output and the reported data. Here, both the reported infections and the reported fatalities are taken into account.

2. MODEL

Following the classical SIR-models introduced by McKendrick [7] and its every-growing number of variants (cf. [8] for an overview), we chose an SEIRD-model to describe the COVID-19 outbreak in Germany. The entire population N is subdivided into five compartments: susceptibles S , exposed E , infected I , recovered R , and deceased D . The virus is transmitted from infected persons to susceptible persons at a time-dependent rate $\beta(t)$ and after an incubation phase of duration κ^{-1} exposed individuals get infectious. Loss of infectivity is gained after γ^{-1} days and with a probability μ , a patient dies from the disease. This leads us to the following five-dimensional ODE system:

$$S' = -\frac{\beta(t)}{N}SI \quad S(t_0) = S_0 = N - E_0 - I_0 - R_0 - D_0 > 0, \quad (1a)$$

$$E' = \frac{\beta(t)}{N}SI - \kappa E \quad E(t_0) = E_0 \geq 0, \quad (1b)$$

$$I' = \kappa E - \gamma I \quad I(t_0) = I_0 > 0, \quad (1c)$$

$$R' = (1 - \mu)\gamma I \quad R(t_0) = R_0 \geq 0, \quad (1d)$$

$$D' = \mu\gamma I \quad D(t_0) = D_0 \geq 0. \quad (1e)$$

The starting point t_0 is chosen as 1 March as on that date number of reported cases exceeded 100 cases for the first time, see Figure 1.

It is immediate to see that the model (1) has non-negative solutions, provided the initial values are all non-negative. Due to the absence of demographic terms, there is just the trivial disease-free equilibrium $S = N$ and $E = I = R = D = 0$. Since the intention of our model is to provide short- and mid-term simulations, we are not interested in its long-term behavior and hence possible endemic equilibria are of no concern.

As a variant of the above basic model, we also consider a delayed differential equation (DDE) version where we introduce a time lag τ between the infected and the deceased state so that the fraction of people who recover or die from the disease is not attained from the amount of infectives on the same day, but from the infectives data τ days earlier. The previous ODE model can thus be seen as a special case of the DDE model with $\tau = 0$.

$$S' = -\frac{\beta(t)}{N}SI \quad S(t_0) = S_0 > 0, \quad (2a)$$

$$E' = \frac{\beta(t)}{N}SI - \kappa E \quad E(t_0) = E_0 \geq 0, \quad (2b)$$

$$I' = \kappa E - \gamma((1 - \mu)I + \mu I(t - \tau)) \quad I(t) = \varphi(t) > 0, \quad (2c)$$

$$R' = (1 - \mu)\gamma I \quad R(t_0) = R_0 \geq 0, \quad (2d)$$

$$D' = \mu\gamma I(t - \tau) \quad D(t_0) = D_0 \geq 0. \quad (2e)$$

Here, $\varphi : [t_0 - \tau, t_0] \rightarrow \mathbb{R}_+$ denotes the initial history of the infected required for the well-posedness of the above delay differential equation. Since the initial value I_0 of the infected at the starting date 1 March is later on subject of the estimation procedure, we assume the initial history to show some exponential behavior

$$\varphi(t) := I_0 \exp\left(-\frac{\ln(0.1)}{\tau}(t - t_0)\right)$$

for $t_0 - \tau \leq t \leq t_0$.

The transmission rate $\beta(t)$ can be related to the Basic Reproduction Number $\mathcal{R}_0$ via

$$\mathcal{R}_0(t) = \frac{\beta(t)}{\gamma}.$$

At the onset of the epidemic, the Basic Reproduction Number $\mathcal{R}_0$ in Germany was estimated to be $\mathcal{R}_0 \simeq 2.4$ —4.1, see [9]. To take the different levels of restriction imposed on the social and economic life, we assume $\beta(t)$ as a step function in time:

$$\beta(t) := \begin{cases} \beta_0, & t < 16 \text{ March} \\ \beta_1, & 16 \text{ March} \leq t < 22 \text{ March} \\ \beta_2, & 22 \text{ March} \leq t < 20 \text{ April} \\ \beta_3, & 20 \text{ April} \leq t \end{cases} \quad (3)$$

Before the first restrictions were imposed on 16 March, the disease was allowed to spread almost uncontrolled. After kindergarden, school and university closings on 16 March, the measures were tightened on 22 March by introducing a contact ban and closing of a large number of shops and businesses. On 20 April, first relaxations were announced and public life began to re-increase, but along with compulsory wearing of masks which has been introduced in late April. For each of these stages we assume an specific contact rate between individuals and hence different transmission rates β_i . The values for the fixed model parameters are given in Table 1.

Table 1: Used parameter values.

Parameter	Value	Unit	Reference
N	83,019,213	–	[10]
κ	1/3	d ⁻¹	[11]
γ	1/10	d ⁻¹	[11]
τ	> 7	d	[11]

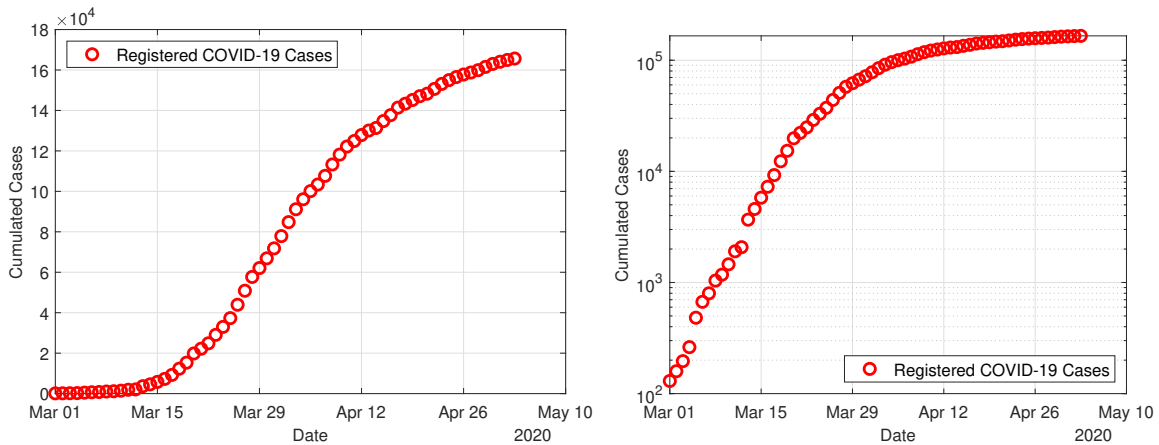


Figure 1: Graphs of cumulative infections in Germany according to Johns Hopkins University from 1 March to 3 May; on the left side with normal scaling and on the right side using semi-logarithmic scaling.

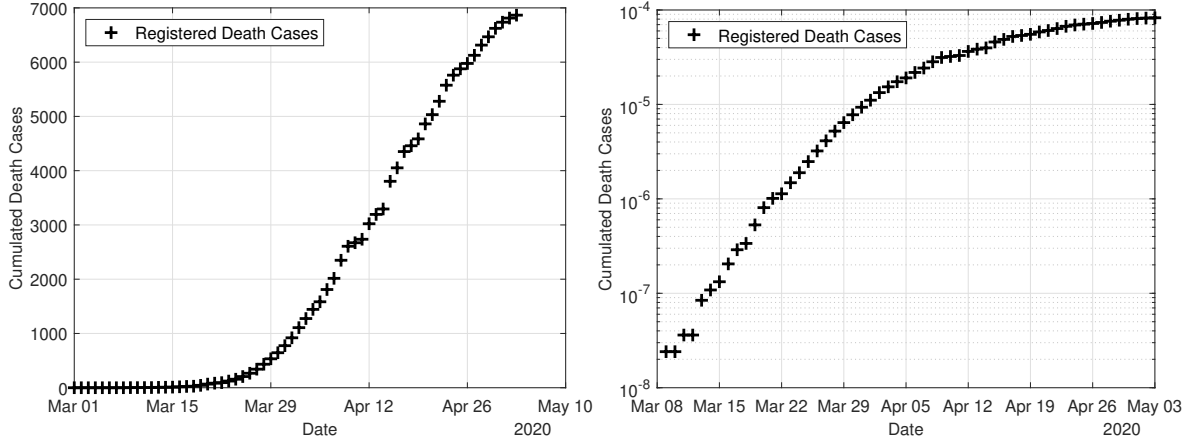


Figure 2: Graphs of cumulative death cases in Germany according to Johns Hopkins University from 1 March to 3 May. The scaling is chosen as in Figure 1.

3. PARAMETER ESTIMATION

The unknown model parameter set u is estimated from a least squares fit of the model output to the given data. Let Y and Z denote the accumulated registered COVID-19 cases or the accumulated COVID-19 deaths in Germany as reported by Johns Hopkins University, see [2]. The reported cases Y consist of the currently infected cases, the recovered and the deceased cases. Since by the very nature of the matter, not all infections are detected, we introduce a detection rate δ . For the currently infected and the recovered ones, we assume that only this proportion δ is tested and detected and hence appears in the statistics; however, we assume no undetected deceased cases. Hence we compare the data Y to $\delta \cdot (I + R) + D$ from the model output. To put special emphasis on the fatalities, we add a term which just compared the reported and the simulated deaths to the cost functional. As a third contribution we add a regularization term proportional to the norm of the estimated parameters to attain a convex function and prevent unrealistic outliers. With this in mind we arrive at the following cost functional:

$$J(u) := \frac{\|\delta(I + R) + D - Y\|_{L^2}^2}{\|Y\|_{L^2}^2} + \frac{\|D - Z\|_{L^2}^2}{\|Z\|_{L^2}^2} + \omega \|u\|_2^2 \quad (4)$$

where $\omega > 0$ denotes some small weight allowing us to adjust the contribution between the normalized least squares terms and the regularization term and $\|f(t)\|_{L^2}^2 = \int_{t_0}^T f(t)^2 dt$ denotes the square of the L^2 -norm of a function f resp. $\|u\|_2^2 = \sum_i u_i^2$ for the square of the Euclidean norm of a vector u .

The parameters to be estimated in model (1) are the transmission rate, the detection rate, lethality and the numbers of exposed on 1 March 2020, i.e.

$$u = (\beta_0, \beta_1, \beta_2, \beta_3, \delta, \mu, E_0) \in \mathbb{R}^7$$

which is the same parameter set as in model (2) with added but fixed time lag τ . For the model with free and to-be-optimized time lag τ , we have the parameter set

$$u = (\beta_0, \beta_1, \beta_2, \beta_3, \delta, \mu, \tau, E_0, I_0) \in \mathbb{R}^9.$$

Here, we also estimate the initial number of infected on 1 March to allow for more flexibility of the model. The optimal parameters u^* are determined by solving the following minimization problem:

$$\min_u J(u) \quad \text{subject to ODE (1) resp. (2),} \quad (5a)$$

$$u^* = \operatorname{argmin}_u J(u). \quad (5b)$$

Table 2 shows the planned simulations including constraints for the optimized parameters in u .

Table 2: Simulations with the respective constraints of the fitted parameters. In Simulation 1 no time lag τ is included in the model. The starting values for I_0 and R_0 are only updated in the first two simulations by division with δ in each iteration. In Simulation 2 the time lag $\tau = 11.5$ is fixed as a mean value within the assumed interval. The parameter τ is also fitted in Simulation 3, just like I_0 . All other unknown parameters in this table are adjusted in each simulation.

Sim.	Model	β_i	δ	μ	τ	E_0	I_0	R_0
1	1	> 0.05	$0.05 - 0.5$	≤ 0.05	0	> 0	$114/\delta$	$16/\delta$
2	2	> 0.05	$0.05 - 0.5$	≤ 0.05	11.5	> 0	$114/\delta$	$16/\delta$
3	2	> 0.05	$0.05 - 0.5$	≤ 0.05	> 7	> 0	> 0	$16/\delta$

Previous investigations in [12] already give us orders of magnitude for the initial values of the optimization for β_i and δ . For the lethality rate μ we assume the upper limit

$$\mu \leq \frac{Z(T)}{R(T)/\delta + Z(T)}$$

whereby $Z(T)$ denotes for the death cases and $R(T)$ denotes the registered recovered individuals at end time T [13]. This upper limit becomes smaller the fewer COVID cases are registered, since δ becomes smaller. For our data set we find

$$\mu \leq \frac{6866}{130600 + 6866} \approx 0.05 \quad (6)$$

based on the registered cases, i.e. this upper limit would match, if $\delta = 1$. Building on the assumption that less than 50% of cases are detected, we also assume a starting value for the lethality rate that is less than half of the calculated upper limit of 5%. The order of magnitude of the time interval between the onset of infectiousness and death is derived from the investigations in [11]. From the timelines available there we derive $\tau \in (7, 17)$. In individual cases this period can be considerably longer, so that τ only represents an average value in the model. The starting values for I_0 and R_0 can be taken from the statistics. Depending on the value of the detection rate, the actual number is calculated by dividing the measured values for the infected and recovered cases by δ . Regarding an estimate of the exposed individuals E_0 at time t_0 , we use a derivation using the Basic Reproduction Number $\mathcal{R}_0$, which indicates how many new infections an infected individual causes on average during its illness in an otherwise susceptible population. In our model, the infected persons I_0 are at different time stages during their infectiousness. As a mean value we assume the middle of this time interval. Thus, up to this point in time they could infect about $I_0 \mathcal{R}_0 / 2$ persons on average. Depending on the assumed Basic Reproduction Number, this results in different starting values for E_0 . The model adaptations are carried out in the simulations with the values $\mathcal{R}_0 \in \{3, 4, 5\}$ and it is checked if significant effects on the other parameters can be found. The selected start values can be seen in Table 3.

Table 3: Orders of magnitude of the initial values for adapting the model to the available data.

param.	β_0	β_1	β_2, β_3	δ	μ	τ	E_0	I_0	R_0
init. val.	0.6	0.4	0.1	0.25	0.02	11.5	$I_0 \mathcal{R}_0 / 2$	$114/\delta$	$16/\delta$

3.1. Adjoint based approach

To solve the minimization problem using adjoint functions we introduce the Lagrangian function

$$\mathcal{L}(u, x, z) = J(u) + \int_{t_0}^T z(t) \cdot \left(g(t, x, u) - \frac{dx}{dt} \right) dt$$

whereby $z = (z_S, z_E, z_I, z_R, z_D)$ denotes the adjoint function regarding the state variable $x = (S, E, I, R, D)$ and $g(t, x, u)$ denotes the right side of the ODE resp. DDE system. It should be noted that within the integral, a scalar product of vectors is calculated. A critical point (u^*, x^*, z^*) needs to fulfill the necessary optimality condition

$$\nabla \mathcal{L}(u^*, x^*, z^*) = 0.$$

For precise details of the following procedure, please refer to [14]. Thus we find the gradient $\nabla_u \mathcal{L}$ regarding the parameters in u

$$\frac{\partial \mathcal{L}}{\partial \beta_i} = 2\omega\beta_i + \frac{1}{N} \int_{t_0}^T \frac{\partial \beta(t)}{\partial \beta_i} SI(z_E - z_S) dt, \quad i = 0, 1, 2, 3 \quad (7a)$$

$$\frac{\partial \mathcal{L}}{\partial \delta} = 2\omega\delta + 2 \int_{t_0}^T (I + R) (\delta(I + R) + D - Y) dt, \quad (7b)$$

$$\frac{\partial \mathcal{L}}{\partial \mu} = 2\omega\mu + \gamma \int_{t_0}^T I(z_D - z_I) dt, \quad (7c)$$

$$\frac{\partial \mathcal{L}}{\partial E_0} = 2\omega E_0 + z_E(t_0) - z_S(t_0), \quad (7d)$$

$$\frac{\partial \mathcal{L}}{\partial I_0} = 2\omega I_0 + z_I(t_0) - z_S(t_0), \quad (7e)$$

resp. in model (2) we obtain, due to the time delay τ ,

$$\frac{\partial \mathcal{L}}{\partial \mu} = 2\omega\mu + \gamma \int_{t_0}^T I(z_I - z_R) + I(t - \tau)(z_D - z_I) dt, \quad (7f)$$

$$\frac{\partial \mathcal{L}}{\partial \tau} = 2\omega\tau + \gamma\mu \int_{t_0}^T (z_I - z_D) \frac{dI}{dt} \Big|_{t=t-\tau} dt. \quad (7g)$$

The adjoint system is given by the equations

$$\frac{dz_S}{dt} = \frac{\beta(t)}{N} I(z_S - z_E), \quad (8a)$$

$$\frac{dz_E}{dt} = \kappa(z_E - z_I), \quad (8b)$$

$$\frac{dz_I}{dt} = \frac{\beta(t)}{N} S(z_S - z_E) + \gamma(z_I - z_R + \mu(z_R - z_D)) - \frac{2\delta(\delta(I + R) + D - Y)}{\|Y\|_{L^2}^2}, \quad (8c)$$

$$\frac{dz_R}{dt} = -\frac{2\delta}{\|Y\|_{L^2}^2} (\delta(I + R) + D - Y), \quad (8d)$$

$$\frac{dz_D}{dt} = -\frac{2}{\|Y\|_{L^2}^2} (\delta(I + R) + D - Y) - \frac{2}{\|Z\|_{L^2}^2} (D - Z), \quad (8e)$$

with the terminal condition $(z_S, z_E, z_I, z_R, z_D)(T) = 0$. By adding the time delay in model (2) we receive

$$\begin{aligned} \frac{dz_I}{dt} = & \frac{\beta(t)}{N} S(z_S - z_E) + (1 - \mu)\gamma(z_I - z_R) - \frac{2\delta}{\|Y\|_{L^2}^2} (\delta(I + R) + D - Y) \\ & + \mu\gamma(z_I(t + \tau) - z_D(t + \tau)) \cdot \chi_{[t_0, T-\tau]}(t). \end{aligned} \quad (8f)$$

Here χ denotes the characteristic function

$$\chi_{[t_0, T-\tau]}(t) = \begin{cases} 1, & t \in [t_0, T - \tau] \\ 0, & \text{else} \end{cases}.$$

Algorithm 1 Pseudocode for the approach including adjoint functions.

```

1:  $u, Y, Z \leftarrow$  load initial values for  $u$  and data
2:  $x, z \leftarrow$  solve ODE resp. DDE for state variable and adjoint function
3:  $J, \nabla J \leftarrow$  compute objective function and gradient regarding  $u$ 
4:  $s \leftarrow$  compute search direction
5: repeat
6:    $J_{old} \leftarrow J$ 
7:    $\vartheta \leftarrow \operatorname{argmin}_{\vartheta > 0} \psi(\vartheta)$  with  $\psi(\vartheta) := J(u + \vartheta s)$ 
8:    $u \leftarrow u + \vartheta s$ 
9:    $x, z, J, \nabla J, s \leftarrow$  update depending on  $u$ 
10: until  $\|J - J_{old}\|_2 < \text{TOL}$ 
11:  $u^*, x^*, z^*, J^* \leftarrow u, x, z, J$ 

```

Algorithm 1 represents the basic framework of the iterative optimization via adjoint functions. To find a preferably global minimum, n multivariate normally distributed start values for u can be created before step 1. These are then tested one after the other with the presented procedure and the best result is chosen. The mean values of this distribution are then the values in Table 3, and the variances can be selected according to the restrictions in Table 2. In step 2 the ODE or DDE are solved using Runge-Kutta methods. Since the state variable is solved forward and the adjoint function backward regarding the time scale due to the initial and end values, this is also called the forward-backward sweep method [14]. In MATLAB the ode45 and dde23 solvers are suitable for this purpose. The search direction s in steps 4 and 9 is selected as Quasi-Newton method (BFGS). Useful alternative search directions are (conjugated) gradient methods [15]. The line search procedure in step 7 cannot be solved analytically in our case. A common method for an appropriate step size ϑ^* would be a backtracking procedure considering the Armijo rule [16]. In the present simulation the procedure in Algorithm 2 is applied. It is based on a Taylor series of $\psi(\vartheta) := J(u + \vartheta s)$ developed around ϑ_0

$$\psi(\vartheta_0 + h) = \psi(\vartheta_0) + \psi'(\vartheta_0)h + \frac{1}{2}\psi''(\vartheta_0)h^2 + \dots$$

where $\psi', \psi'', \dots$ stand for the respective derivatives of ψ regarding ϑ . Based on this, we assume that ψ for $\vartheta_0 = 0$ and sufficiently small values for $h = \vartheta$ can be approximated by a parabola with

$$\begin{aligned}\psi(\vartheta) &\simeq a\vartheta^2 + b\vartheta + c \\ \psi'(\vartheta) &\simeq 2a\vartheta + b.\end{aligned}\tag{9}$$

Using the information $\psi(0) = J(u)$ and $\psi'(0) = \nabla J(u) \cdot s$ associated with a calculated value $\psi(\vartheta_1) = J(u + \vartheta_1 s)$ for small and fixed ϑ_1 allows to derive the parameters

$$\begin{aligned}c &= \psi(0), \\ b &= \psi'(0), \\ a &= (\psi(\vartheta_1) - \psi'(0)\vartheta_1 - \psi(0))/\vartheta_1^2,\end{aligned}$$

and, by using the necessary condition $\psi'(\vartheta^*) = 0$, find the optimum of the parabola in (9)

$$\vartheta^* = -b/(2a) = -0.5\psi'(0)\vartheta_1^2/(\psi(\vartheta_1) - \psi'(0)\vartheta_1 - \psi(0)).$$

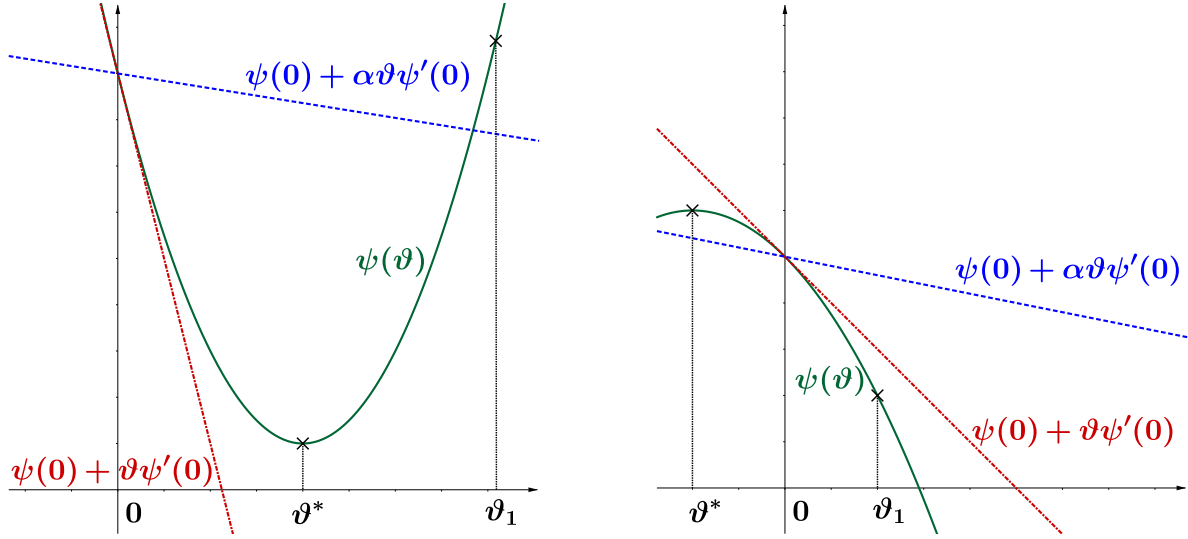


Figure 3: Graphical example to approximate the optimal value for ϑ^* with parabola linesearch. The left figure shows that the Armijo rule $\psi(\vartheta) \leq \psi(0) + \alpha\vartheta\psi'(0)$ is not fulfilled for ϑ_1 and the new step size is determined using the parabola minimum ϑ^* . To make sure that the possible minimum of the parabola is below that line, one chooses a small value for $\alpha \in (0, 0.5)$, e.g. $\alpha = 1e-4$. In the right figure the Armijo rule is already fulfilled with the fixed increment ϑ_1 which can be adopted. There can also be a parabola maximum, so that ϑ^* takes a negative value. However, this is circumvented because in this case, there is no optimization of the step size.

Algorithm 2 Pseudocode for line search in step 7 of Algorithm 1.

```

1:  $u, J(u), \nabla J(u), s \leftarrow \text{input}$ 
2:  $\vartheta \leftarrow 1$ 
3:  $\psi(0) \leftarrow J(u)$ 
4:  $x \leftarrow \text{compute state variable depending on } u + \vartheta s$ 
5:  $\psi(\vartheta) \leftarrow J(u + \vartheta s)$ 
6:  $\psi'(0) \leftarrow \nabla J(u) \cdot s$ 
7:  $\alpha \leftarrow \text{value in } (0, 0.5)$ 
8: if  $\psi(\vartheta) > \psi(0) + \alpha\vartheta\psi'(0)$  then
9:   repeat
10:     $\vartheta \leftarrow -0.5\psi'(0)\vartheta^2 / (\psi(\vartheta) - \psi'(0)\vartheta - \psi(0))$ 
11:     $x \leftarrow \text{update depending on } u + \vartheta s$ 
12:     $\psi(\vartheta) \leftarrow J(u + \vartheta s)$ 
13:  until  $\psi(\vartheta) \leq \psi(0) + \alpha\vartheta\psi'(0)$  (Armijo rule)
14: end if
15:  $\vartheta^* = \vartheta$ 

```

The effect of the weight ω can be seen on the diagonal of the Hessian matrix in model (1)

$$\nabla_u^2 \mathcal{L} = 2 \text{diag} \left(\omega, \omega, \omega, \omega, \omega + \int_{t_0}^T (I + R)^2 dt, \omega, \omega, \omega \right)$$

whereby all other entries in $\nabla_u^2 \mathcal{L}$ are 0. The value of ω directly influences the definiteness of the Hessian matrix and thus the convexity of the objective function. For this reason, different values for ω are tested in the simulations.

3.2. Metropolis algorithm

According to the procedure described in [17], a Metropolis algorithm (cf. [18], [19], [20]) for model (2) can be set up using the initial history and initial values for the to-be-estimated parameter set u . Using the parameter set u_0 as of Table 3 as starting conditions, we assign random draws u_{new} from a normally distributed (and thus symmetric) proposal function q , i.e. $u_{new} \sim q(u_{new}|u_{i-1})$, in every iteration i .

Using the previously defined $J(u)$ as the target distribution, we calculate the approximative distribution by

$$\pi(u) = c \cdot \exp\left(-\frac{J(u)^2}{2\sigma^2}\right), \quad (10)$$

whereby c is an arbitrary value in $\mathbb{R}$. For the acceptance probability, it follows

$$\alpha(u_{new}|u_{i-1}) = \min\left\{1, \frac{\pi(u_{new}) \cdot q(u_{i-1}|u_i)}{\pi(u_i) \cdot q(u_i|u_{i-1})}\right\} = \min\left\{1, \frac{\pi(u_{new})}{\pi(u_i)}\right\}. \quad (11)$$

In Eq.(11) we can see that the value of c is redundant as it cancels out in the division.

If the sample is accepted with the probability α , we set $u_i = u_{new}$; with the probability $1 - \alpha$, the sample is declined, meaning $u = u_{i-1}$ [21], [17].

Algorithm 3 Pseudocode for the Metropolis algorithm.

- 1: $u, Y, Z \leftarrow$ load initial values for u and data
 - 2: $x, z \leftarrow$ solve ODE resp. DDE for state variable
 - 3: $J \leftarrow$ compute objective function regarding u
 - 4: $\sigma \leftarrow$ standard distribution of the solution, i.e. $I + R + D$ over time
 - 5: $s \leftarrow$ set step size (standard deviation) for the algorithm, e.g. $s := u/100$
 - 6: **repeat**
 - 7: $u_{old} \leftarrow u$ from previous draw
 - 8: $\hat{u}_{new} \leftarrow u \sim \mathcal{N}(u_{old}, s)$
 - 9: $x, z, J(\hat{u}_{new}) \leftarrow$ update depending on u
 - 10: $\alpha \leftarrow \min\left\{1, \exp\left(\frac{J(u_{old})^2 - J(u_{new})^2}{2\sigma^2}\right)\right\}$
 - 11: $u_{new} \leftarrow \hat{u}_{new}$ with probability α and $u_{new} := u$ with probability $1 - \alpha$
 - 12: **until** maximum value of draws is reached
 - 13: $u^*, x^*, J^* \leftarrow$ means of all u, x, J
-

Algorithm 3 represents the basic framework of the iterative optimization via the Metropolis algorithm. In step 1, the mean values of this distribution as of Table 3 are loaded as well as the variances according to the restrictions in Table 2. In step 2 the ODE or DDE are again solved using Runge–Kutta methods via MATLAB’s ode45 and dde23 solvers. The step size s in step 5 is selected as a fraction of the initial guess for the parameter set u so that the parameters are allowed move with an individual “speed” through the search space. In steps 6 to 12, the process is repeated for all draws, the number of draws in our case is set to $2e+4$. Alternatively, you can think about termination conditions, but we avoided this due to the random nature of the system. Firstly, the update of the parameter set u is done by taking a random value out of the normal distribution with mean u and standard deviation s . After solving the system in step 9, the cost functional $J(u)$ is compared to the previous cost functional with the function α in step 10 and the new parameter set is accepted or rejected according to 11 in step 11. The estimation parameter set can then be computed out of the mean value of the draws in step 13. Alternatively, in case of non–convergence, you can compute the best fitting u of the set and use this as initial value as of step 1 again, to attain better results.

Choosing the weights ω for the target function $J(u)$ was done under two purposes. The first purpose was to create a convex target function so that the algorithm does not converge to local minima (see also the previous subsection for this). The Metropolis algorithm allows steps into parameter sets having a “worse” target distribution with a certain probability, but it is still possible that it runs into local but not global minima after a *final* amount of steps which justifies the usage of the term $\omega \|u\|^2$. The other purpose is to not have a too large ω so that the model-related terms still have a major impact on the outcome of $J(u)$. For these two

regards, we found that a range for ω between $\omega := 10^{-9}$ and $\omega := 10^{-7}$ is decent, but we will also present the results if we neglect the term with ω , i.e. $\omega = 0$. For values $\omega \in (0, 10^{-9})$ no significant changes in the outcomes to $\omega = 0$ were detected, while for $\omega > 10^{-7}$ the model-related terms are negligible and the results are quite unrealistic.

4. NUMERICAL RESULTS AND COMPARISON OF THE ALGORITHMS

Table 4: Numerical Results.

	Algorithm Simulation	Adjoint			Metropolis		
		1	2	3	1	2	3
β_0		0.60	0.64	0.62	0.55	0.70	0.64
β_1		0.50	0.48	0.51	0.49	0.40	0.64
β_2		0.101	0.082	0.092	0.113	0.085	0.086
β_3		0.099	0.050	0.058	0.054	0.055	0.055
δ		0.31	0.27	0.18	0.29	0.20	0.19
μ		0.015	0.018	0.011	0.013	0.013	0.011
τ		0	11.5	9.0	0	11.5	7.3
$E_0 + I_0 + R_0$		831	1,105	1,512	1,255	854	1,090
$(J(u) - \omega \ u\ _2^2) \cdot 10^3$		23.0	9.1	6.1	18.1	8.2	3.2
Iterations		23	22	31	20000	20000	20000

Table 4 shows the respectively best numerical results of the two algorithms. The values for the transmission parameters β_i are of similar magnitudes in almost all simulations and algorithms. In isolated cases there are more significant deviations, such as $\beta_1 = 0.64$ in Simulation 3 of the Metropolis approach or the value $\beta_3 = 0.099$ in Simulation 1 of the adjoint approach. The values show that the dynamics of the model at the beginning of the measurement period with $\beta_0 \simeq 0.6$ suggest a much higher $\mathcal{R}_0$ than assumed. The first measures lead to a small to moderate reduction of the transmission rate to $\beta_1 \simeq 0.5$, whereas the following lockdown causes a significant decrease of the transmission rate to $\beta_2 \simeq 0.1$. This also fits with the estimates of the RKI that the Basic Reproduction Number is said to have dropped to a value of around $\mathcal{R}_0 \simeq 1$ due to the extensive restrictions [3]. In the last phase of the data adaptation the transfer rate drops to $\beta_3 \simeq 0.06$. Here, due to the loosening of the measurements, one would expect an increase of the transmission rate. However, these were introduced very slowly and under very strict hygiene measures, combined with a mask requirement in public spaces, which apparently has decreased the β value. Regarding the detection rate δ we find values of around 20 – 30% in all cases. This means that according to the simulations, the actual number of infected people is 3–5 times higher than the official reports. The computed lethality is between 1–2% and is therefore roughly a third of 5% which was calculated in (6) regarding the registered cases at the end time point T . The average time interval τ between the onset of infectivity and death in Simulation 3 is between 7 and 10 days. The influence of τ is also evident with regard to the normalized least squares terms $J(u) - \omega \|u\|_2^2$. By adding a fixed time lag in Simulation 2 and then adjusting it in the third simulation, a significant improvement is shown in all algorithms as $J(u)$ is considerably smaller. Regarding the magnitudes of the least-squares terms, the algorithms show similar values in comparison to each other and lead to useful adjustments with minor deviations of the model from the available data sets. This is also illustrated by the graphical results which are shown in Appendices A and B. The sum of the initial values $E_0 + I_0 + R_0$ lies within a realistic range at $\simeq 1000$. Thus, the unknown initial value for the exposed individuals E_0 is approximately in the order of magnitude of the infected I_0 with an upward tendency, as expected. The variation regarding the initial value for $E_0 = I_0 \mathcal{R}_0 / 2$ in the optimization does not lead to significant differences in the results when $\mathcal{R}_0 \in \{3, 4, 5\}$ is changed. For this reason, the results are presented here only for initial estimations of $\mathcal{R}_0 = 3$. In the case of the Metropolis algorithm, the number of iterations is much higher than in the adjoint approach. This is due to the fact that the Metropolis approach relies on random draws and thus a large amount of draws is needed to obtain convergence and to diminish the effect of outliers. This seemingly disadvantageous property of the Metropolis algorithm is partly counter-balanced when using n multivariate normally distributed values for u as starting guesses for the adjoint-based optimization. This also increases the iteration number by a factor n . On the other hand, this would have the consequence that the probability of reaching a global minimum for $J(u)$ would increase significantly. This aspect is already been cared for in the Metropolis algorithm so no additional computations are required unless the chain statistics (as to be

seen in the following sections). The value for $J(u)$, especially in Simulation 3 are slightly more accurate using the Metropolis algorithm. The comparison of the runtimes in Simulation 3 on an Intel Core i5–6400 with 2.7 GHz and 16 MB–RAM also reflects this. Due to the higher number of iterations, the Metropolis algorithm also has a longer runtime, see Table 5.

Table 5: Average required runtime of the algorithms on an Intel i5–6400 with 2.7 GHz and 16 MB–RAM.

Algorithm	Average runtime [s]
Adjoint approach	10
Metropolis	140

Additionally, the influence of the weight ω on the optimization is tested. Table 6 shows the results of the least squares term $J(u) - \omega \|u\|_2^2$ for Simulation 3 with the two algorithms and different weights.

Table 6: Values for the normalized least squares terms $(J(u) - \omega \|u\|_2^2) \cdot 10^3$ for the optimization with different weights ω regarding the algorithms in Simulation 3.

Algorithm	$\omega = 0$	$\omega = 10^{-9}$	$\omega = 10^{-8}$	$\omega = 10^{-7}$
Adjoint approach	8.9	8.8	6.1	12.0
Metropolis	3.8	3.2	3.4	4.1

The results show that an appropriate weight value is $\omega \simeq 10^{-8}$ resp. 10^{-9} , depending on the chosen algorithm. If the weight is too large, the value of the least squares term also deteriorates. This makes sense since the disturbance caused by $\omega \|u\|_2^2$ on the objective function becomes too large. On the other hand, however, a sufficiently small value for ω leads to better optimization performance, since a weight of $\omega = 0$ on the other hand gives a worse result.

4.1. Specific results for the adjoint approach

As shown in Table 4, the approach with adjoint functions leads to similar numerical results as the other tested routine. The graphical results of Simulation 3 are shown in Figure 4.

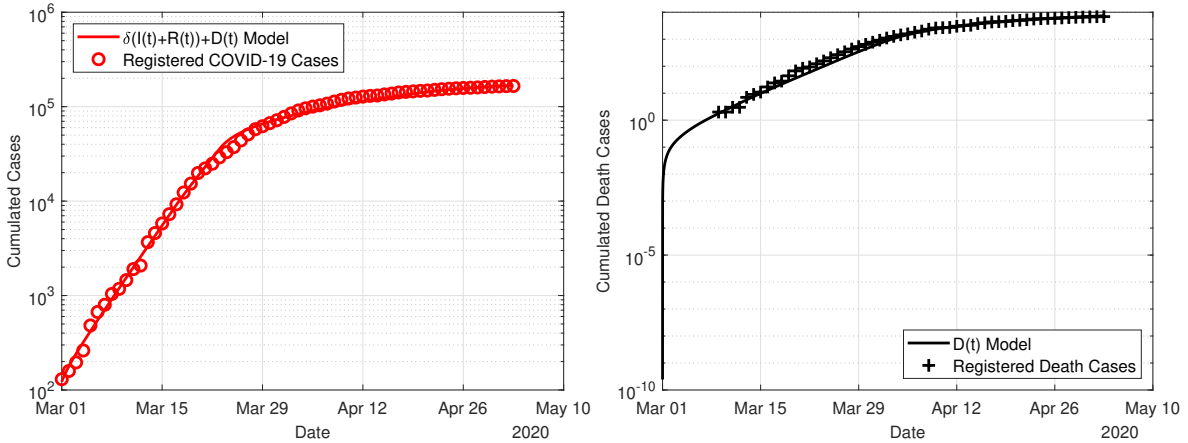


Figure 4: Plots for $\tau := \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-8}$.

The necessary number of iterations until the convergence of the algorithm shows that the algorithm moves quickly to the corresponding minima, see Figure 5. The process clarifies that the algorithm is very close to the optimal objective function value already after 15 iterations and needs the remaining calculation steps to

reach the given tolerance limit $TOL = 10^{-12}$. However, the prerequisite for rapid convergence is a good starting value for u .

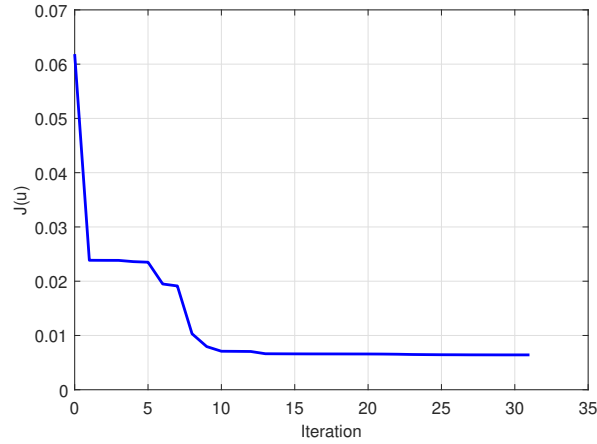


Figure 5: Development of the objective function J depending on the corresponding iteration step.

In addition to the presented simulations with restrictions, the algorithm was performed without limitations for the searched parameters, see Table 7 and Figure 6.

Table 7: Numerical results of Simulation 3 without restrictions concerning the estimated parameters.

β_0	β_1	β_2	β_3	δ	μ	τ	$E_0 + I_0 + R_0$	$J(u) - \ u\ _2^2$
0.77	0.46	0.27	0.41	0.002	0.0001	7	65046	$7 \cdot 10^{-4}$

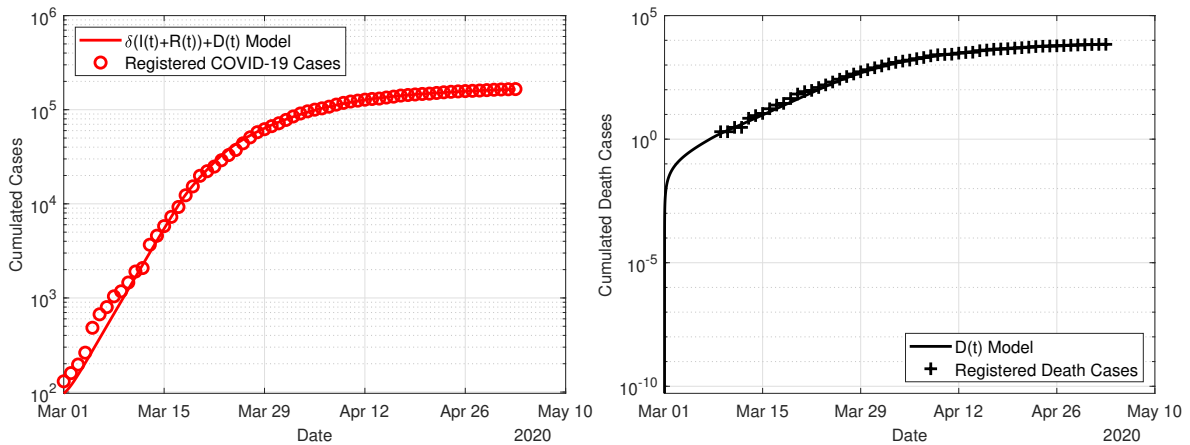


Figure 6: Graphical results of Simulation 3 without restrictions concerning the estimated parameters.

The results show that the normalized least squares term $J(u) - \|u\|_2^2$ can be reduced significantly compared to the restricted variants. It is noticeable, however, that the fitted value for the detection rate δ is very small at about 0.02%. This would mean that only every 500th infected person would be registered. This seems unrealistic, even if the dark figure is unknown. The values for transmission rate, lethality and actual number of exposed, infected and recovered at the beginning of the measurement period are changed accordingly.

Due to the very low detection rate in this simulation, the spread of the disease would have been much more intense than expected.

4.2. Specific results for the Metropolis algorithm approach

We now consider the value for

$$J(u) - \omega \|u\|_2^2 = \frac{\|\delta(I + R) + D - Y\|_{L^2}^2}{\|Y\|_{L^2}^2} + \frac{\|D - Z\|_{L^2}^2}{\|Z\|_{L^2}^2},$$

i.e. the cost functional $J(u)$ without the last term including the weight ω . This way we can compare the simulations with different weights ω in terms of $J(u)$ because the last term trivially raises along with ω .

Table 8: $(J(u) - \omega \|u\|_2^2) \cdot 10^3$ for the different weights ω .

Simulation	$\omega = 0$	$\omega = 10^{-9}$	$\omega = 10^{-8}$	$\omega = 10^{-7}$
1	18.6	18.1	18.6	21.7
2	8.7	8.2	9.2	9.6
3	3.8	3.3	3.4	4.1

Table 8 shows that the weight $\omega = 10^{-9}$ always yields the best, i.e. smallest values for the given cost functional $J(u)$. Moreover, what you can also see in Tables 10, 12 and 14 in Appendix B, the value $J(u)$ for the weight $\omega = 10^{-9}$ is larger than the value $J(u)$ with the weight $\omega = 0$, even when the term $10^{-9} \cdot \|u\|_2^2$ is *not* subtracted, which means that interestingly, the simulation with $\omega = 10^{-9}$ provides a better result for a different cost functional.

The plots for the infected and dead cases in Simulation 3 with $\omega = 1e-9$, thus the best simulation, are shown in Figure 7.

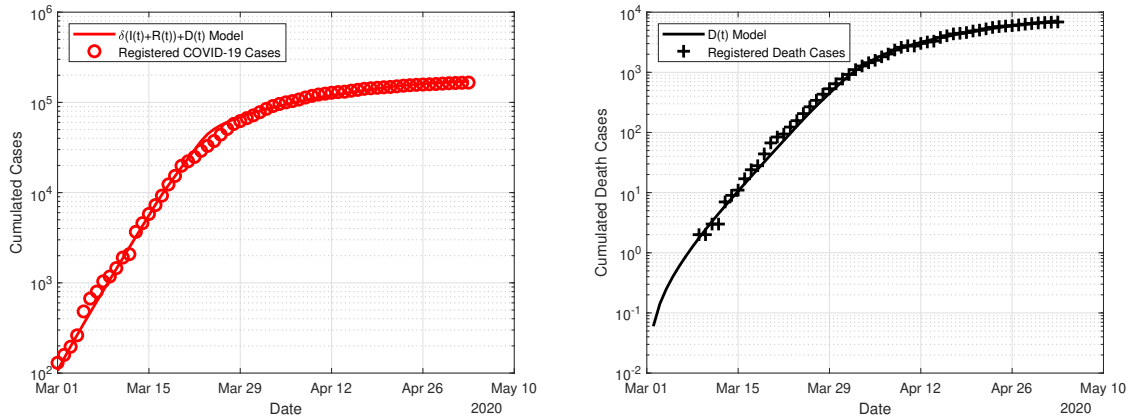


Figure 7: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 := 3$ and $\omega = 10^{-9}$.

The chain statistics done with the optimal results in Simulation 3 for $\omega = 10^{-9}$ as of Figure 8 show that for most parameters a normal distribution is visible and thus the Metropolis algorithm appears to have converged. The parameter τ does not appear to be normally distributed, but still remains in the range from 7–8 days. This also affects some smaller side peaks regarding the other parameters. As the infection data has the step size of 1 day, we assume that no further optimization within that range is possible, so an estimation of $\tau \approx 7\text{--}8$ days is decent enough.

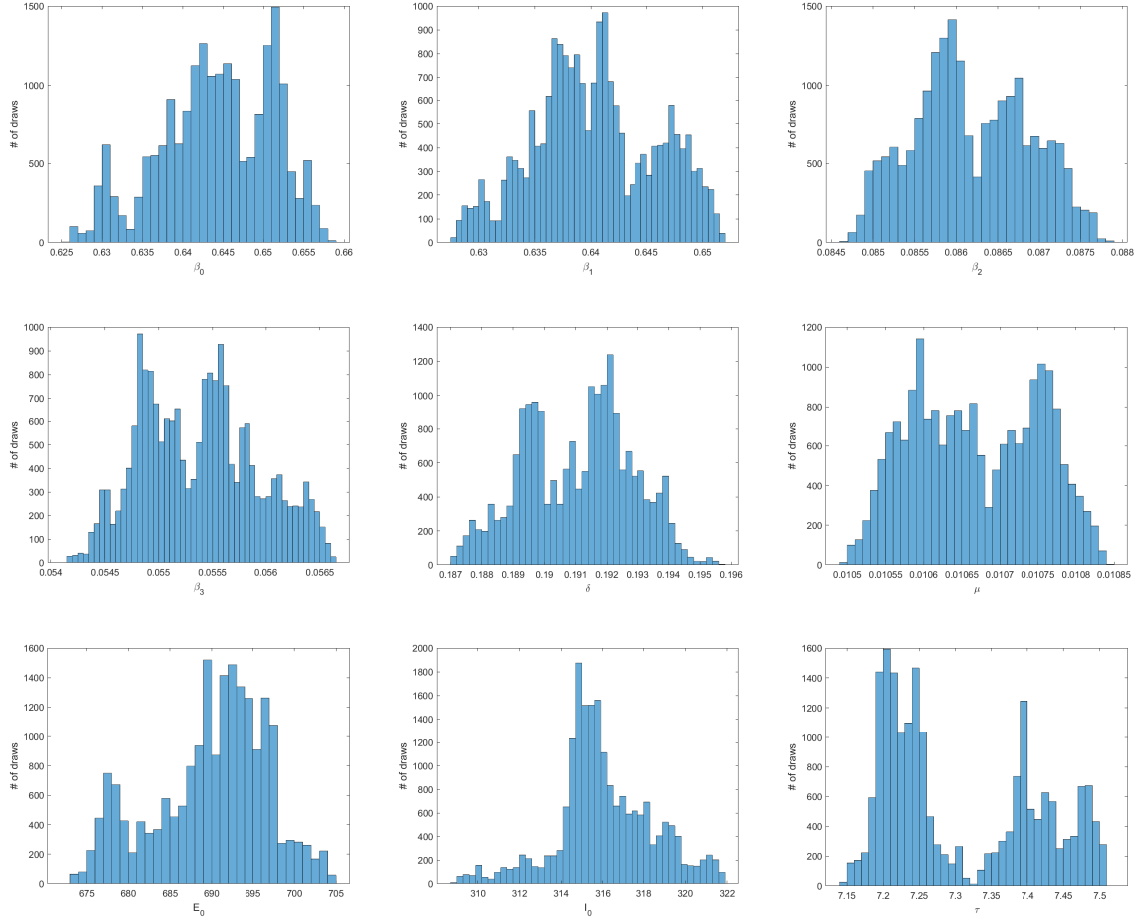


Figure 8: Parameter statistics for Simulation 2 and $\omega = 10^{-9}$, using the best approximation with respect to $J(u) - \omega \|u\|_2^2$ as starting value and a step size of $u_0/1000$. Except of τ , most histograms appear roughly normally distributed around their mean values.

A detailed numerical analysis as well as figures for all relevant plots can be found in Appendix B. In the figures it is also visible that with fixed values $\tau = 0$ or $\tau = 11.5$ the estimated death cases run after resp. run ahead of the data.

5. CONCLUSION

In the present work, two SEIRD-models for modelling the COVID-19 outbreak in Germany were adapted to existing data from 1 March to 3 May. Two different approaches for the estimation of parameters and approximation of the infection data were used and their results and performance were compared. Regarding the graphical and numerical results, all routines have provided similar meaningful results. Each approach has advantages and disadvantages and should be selected depending on the application needs, time, possible analytical and programming effort. The Corona outbreak results show that the restrictions taken by the authorities have had a major impact on the dynamics of spread. The Basic Reproduction Number could be reduced from a presumably much higher value than the assumed $\mathcal{R}_0 \simeq 3$ to the epidemiologically important limit $\mathcal{R}_0 \simeq 1$. Adding a time lag τ between the onset of infectiousness and death significantly increases the accuracy of the tested model. This time delay is estimated by the data adjustment to an average of 8

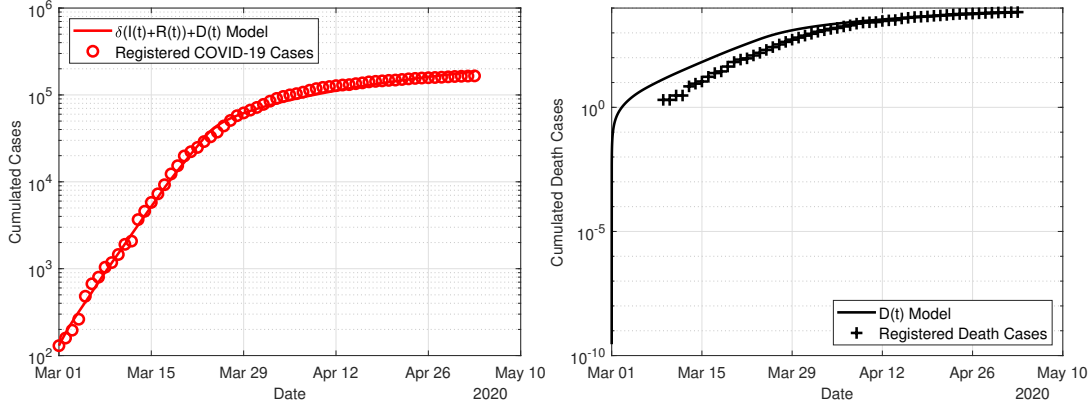
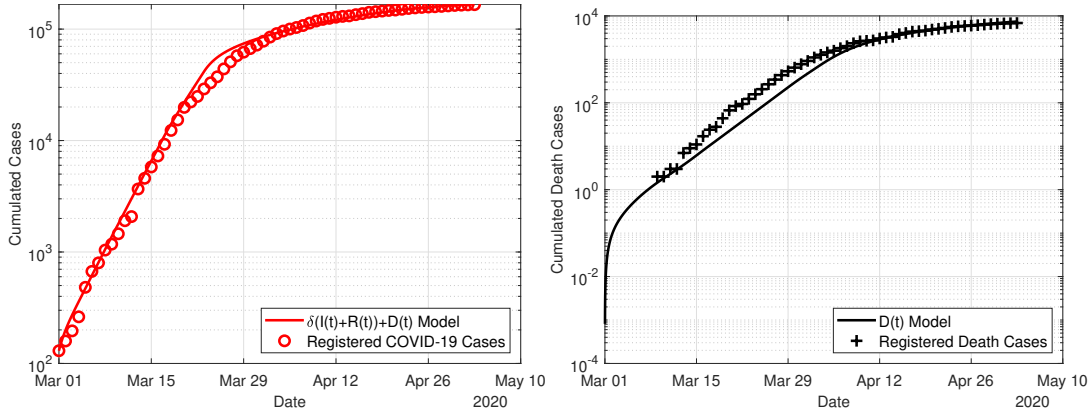
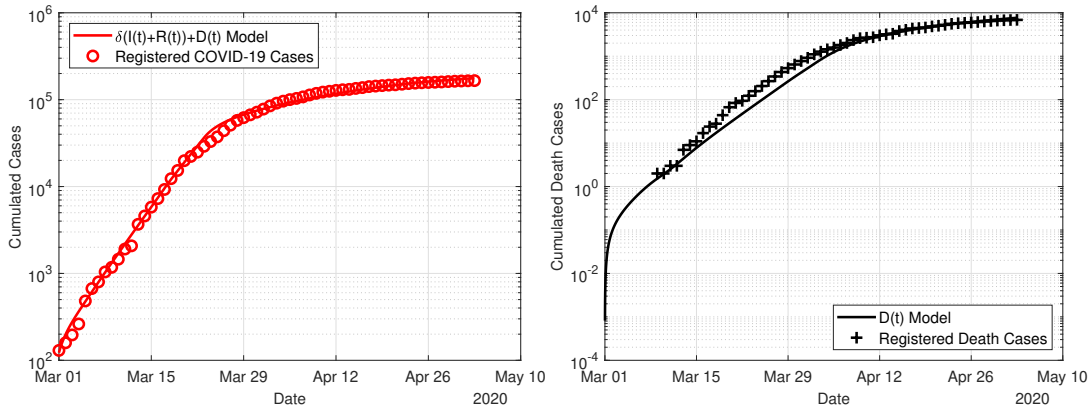
days, although in reality there may be very different values depending on how long life-support measures are maintained in intensive care units. The adjustment regarding the detection rate and lethality showed that, according to the model, the actual number of infected people is approximately 3–5 times higher than registered and at $\mu \approx 1\text{--}2\%$, the lethality is lower than assumed.

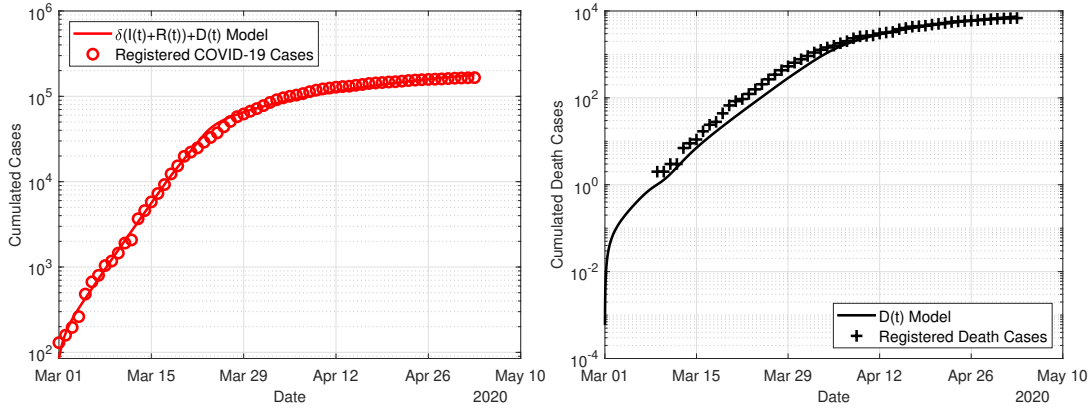
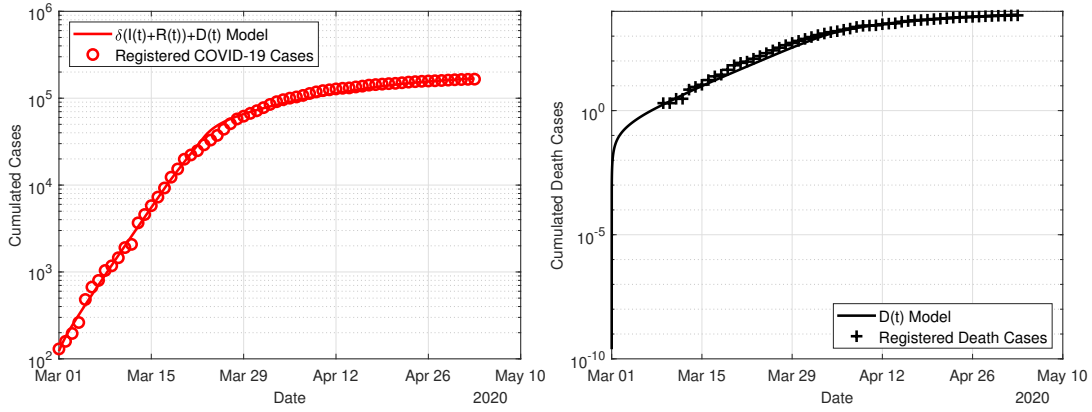
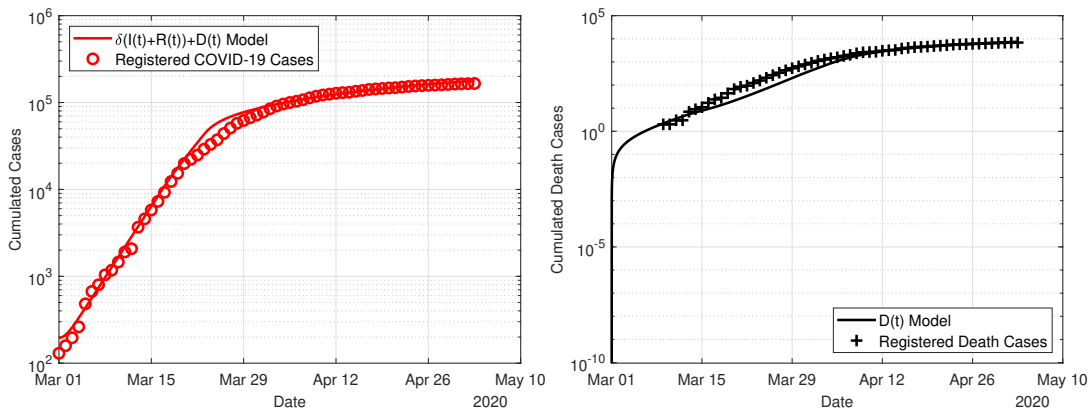
Conceivable extensions of the present work would be the application to other countries, the integration of travel or commuting after the relaxation of exit restrictions or the integration of control variables to mathematically derive the optimal time intervals for future lockdowns. With respect to the latter, in order to detect a new increase in infections early on – before it returns to exponential growth – a measure within the model of the possible increase in transmission rate is required.

REFERENCES

- [1] World Health Organization (WHO), Novel Coronavirus (2019-nCoV) situation reports. www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports, last visited: 18 June 2020
- [2] Johns Hopkins University, Time series of confirmed COVID-19 cases globally, github.com/CSSEGISandData/COVID-19/blob/master/csse_COVID_19_data/csse_COVID_19_time_series/time_series_COVID19_confirmed_global.csv, last visited: 18 June 2020.
- [3] Robert-Koch-Institute, Daily situation reports. www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Situationsberichte/Gesamt.html, last visited: 18 June 2020.
- [4] Federal Government of Germany, Guidelines for reducing social contacts. www.bundesregierung.de/breg-de/themen/coronavirus/besprechung-der-bundeskanzlerin-mit-den-regierungschefinnen-und-regierungschefs-der-laender-1733248 (in German), last visited: 18 June 2020.
- [5] Federal Government of Germany, Contact Restrictions Extended. <https://www.bundesregierung.de/breg-en/news/fahrplan-corona-pandemie-1744276>, last visited: 18 June 2020.
- [6] Chambers, M., Carrel, P., Germany eases lockdown, with 'emergency brake' on hand if needed. <https://www.reuters.com/article/us-health-coronavirus-merkel-idUSKBN22I24E>, Accessed on May 6, 2020 last visited: 18 June 2020.
- [7] Kermack, W.O., McKendrick, A.G., Contributions to the mathematical theory of epidemics—I, *Bltn. Mathcal. Biology.*, 53, 33–55, 1991.
- [8] Martcheva, M., *An introduction to mathematical epidemiology*, Springer, 2015.
- [9] Read, J. M., Bridgen, J.R.E., Cummings, D.A.T., Ho, A., Jewell, C.P., Novel coronavirus 2019-nCoV: early estimation of epidemiological parameters and epidemic predictions. medRxiv www.medrxiv.org/content/10.1101/2020.01.23.20018549v2, last visited: 18 June 2020.
- [10] Statistisches Bundesamt (Germany), Bevölkerungsstand (31.12.2018). https://www.destatis.de/DE/Home/_inhalt.html, 2020.
- [11] Robert-Koch-Institute, Modellierung von Beispielszenarien der SARS-CoV-2-Epidemie 2020 in Deutschland. https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Modellierung_Deutschland.pdf?__blob=publicationFile, last visited: 22 June 2020.
- [12] Götz, T., Heidrich, P., Novel Coronavirus (2019-nCoV) situation reports. doi: <https://doi.org/10.1101/2020.04.23.20076992>
- [13] Robert-Koch-Institute, Corona fact sheet. https://www.rki.de/DE/Content/InfAZ/N/Neuartiges_Coronavirus/Steckbrief.html, last visited: 18 June 2020.
- [14] Lenhart, S., Workman, J.T., *Optimal control applied to biological models.*, CRC Press, 2007.
- [15] Nocedal, J., Wright, S., *Numerical Optimization*, Springer, 2006.
- [16] Armijo, L., Minimization of functions having Lipschitz continuous first partial derivatives, *Pacific J. Math.*, 16 (1), 1–3, 1966.
- [17] Schäfer, M., Götz, T., Modelling Dengue Fever Epidemics in Jakarta, *Int. J. Appl. Comput. Math.*, 6, 2020.
- [18] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.W., Teller, A.H., Teller, E., Equation of State Calculations by Fast Computing Machines, *J.Chem. Phys.*, 21, 1087–1092, 1953.
- [19] Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B., *Bayesian Data Analysis*, 2nd Edition, Chapman and Hall, London, 1996.
- [20] Gilks, W.R., Richardson, S., Spiegelhalter, D.J., *Markov chain Monte Carlo in Practice*, Chapman and Hall/CRC, 1996.
- [21] Rusatsi, D.N., *Bayesian analysis of SEIR epidemic models*, Dissertation, Lappeenranta University of Technology. <http://lutpub.lut.fi/>, last visited: 18 June 2020.

APPENDIX A – PLOTS FOR THE ADJOINT APPROACH

Figure 9: Plots for $\tau = 0$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-8}$.Figure 10: Plots for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-8}$.Figure 11: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 0$.

Figure 12: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-9}$.Figure 13: Plots for $\tau := \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-8}$.Figure 14: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-7}$.

APPENDIX B – RESULTS AND PLOTS FOR METROPOLIS ALGORITHM

Simulation 1 – No delay and fixed initial infectives

Table 9: Estimates for $\tau = 0$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $R_0 = 16/\delta$, $\mathcal{R}_0 = 3$ after $r = 20000$ draws and using a step size of $s = u_0/100$.

Parameter	$\omega = 0$		$\omega = 10^{-9}$		$\omega = 10^{-8}$		$\omega = 10^{-7}$	
	mean	std.	mean	std.	mean	std.	mean	std.
β_1	.5822	.0353	.5525	.0439	.5935	.0177	.6381	.0227
β_2	.5378	.0169	.4936	.0350	.4828	.0160	.4645	.0348
β_3	.1140	.0111	.1130	.0067	.1094	.0048	.1014	.0130
β_4	.0671	.0032	.0538	.0033	.0502	.0027	.0510	.0056
δ	.2307	.0089	.2933	.0116	.2137	.0104	.3142	.0309
μ	.0105	.0010	.0131	.0016	.0095	.0007	.0137	.0011
E_0	540.7	22.5	811.4	41.5	819.8	52.9	440.8	16.1

Table 10: $J(u) \cdot 1000$ for the different weights in Simulation 1. The column represents the weight that is used for $J(u)$ in the Metropolis algorithm and the row shows the value of $J(u)$ for all four ω .

w.r.t. ω	weight ω			
	0	10^{-9}	10^{-8}	10^{-7}
0	18.6	18.1	18.6	21.7
10^{-9}	19.2	18.9	19.5	22.1
10^{-8}	24.0	26.2	28.1	25.0
10^{-7}	72.3	99.1	114.2	54.3

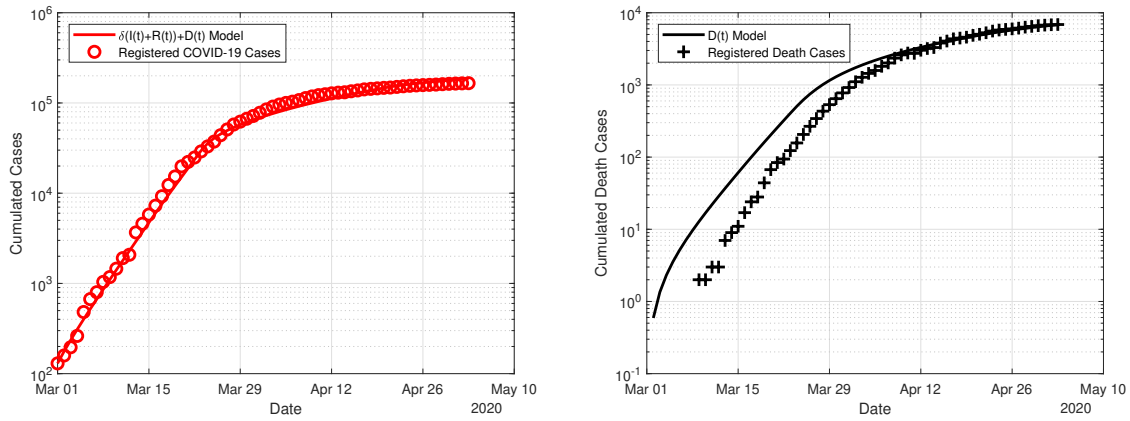
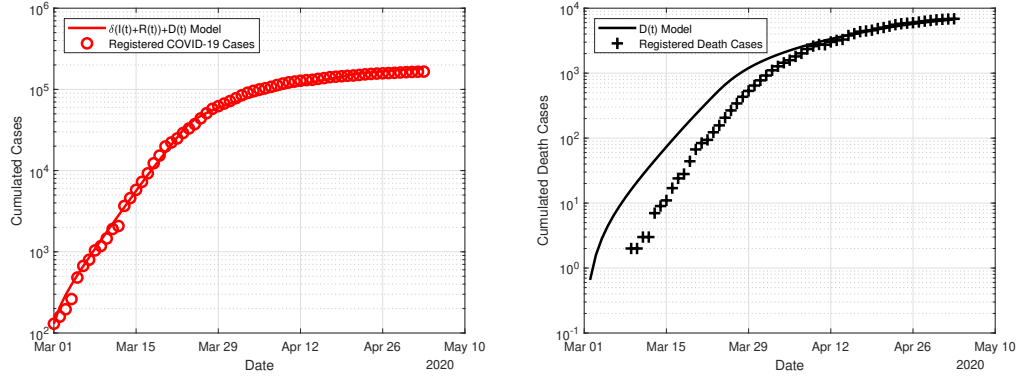
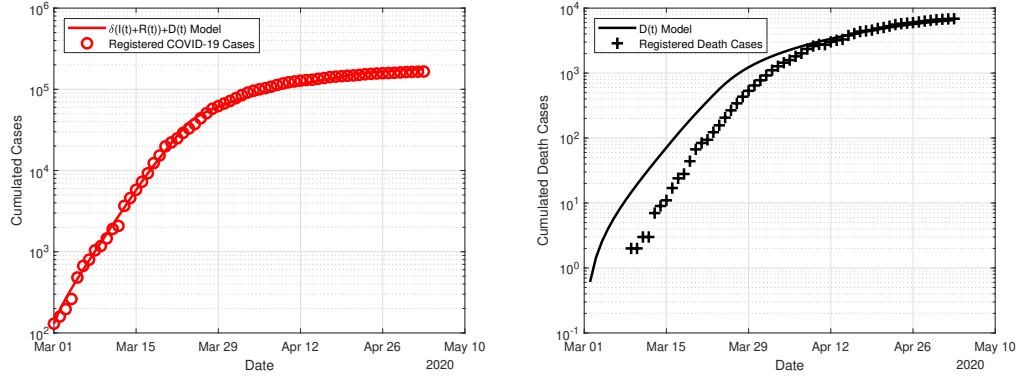
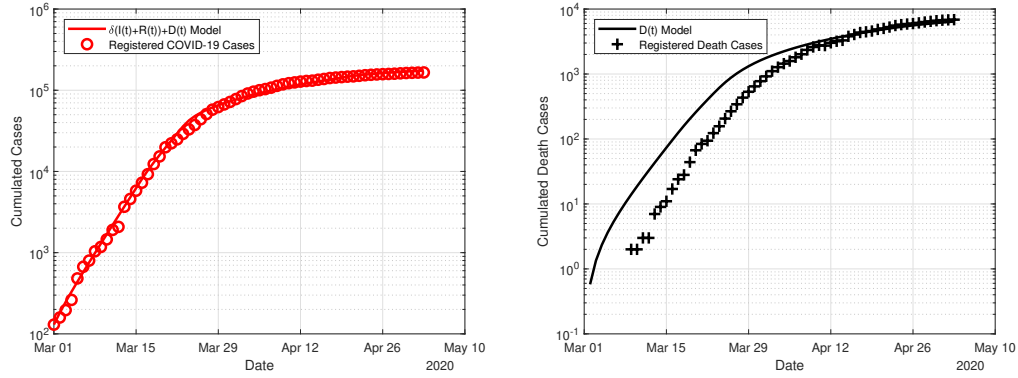


Figure 15: Plots for $\tau = 0$, $E_0 = \text{free}$, $I_0 = \frac{114}{\delta}$, $\mathcal{R}_0 = 3$ and $\omega = 0$.

Figure 16: Plots for $\tau = 0$, $E_0 = \text{free}$, $I_0 = \frac{114}{\delta}$, $\mathcal{R}_0 = 3$ and $\omega = 0$.Figure 17: Plots for $\tau = 0$, $E_0 = \text{free}$, $I_0 = \frac{114}{\delta}$, $\mathcal{R}_0 = 3$ and $\omega = 0$.Figure 18: Plots for $\tau = 0$, $E_0 = \text{free}$, $I_0 = \frac{114}{\delta}$, $\mathcal{R}_0 = 3$ and $\omega = 0$.

Simulation 2 – Fixed delay and initial infectives

Table 11: Estimates for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $R_0 = 16/\delta$, $\mathcal{R}_0 = 3$ after $r = 20000$ draws and using a step size of $s = u_0/100$.

Parameter	$\omega = 0$		$\omega = 10^{-9}$		$\omega = 10^{-8}$		$\omega = 10^{-7}$	
	mean	std.	mean	std.	mean	std.	mean	std.
β_1	.6735	.0538	.7045	.0600	.6391	.0411	.6678	.0508
β_2	.4414	.0250	.3951	.0336	.4823	.0323	.5011	.0323
β_3	.0810	.0073	.0846	.0075	.0820	.0059	.0790	.0090
β_4	.0672	.0042	.0552	.0073	.0520	.0027	.0605	.0091
δ	.2055	.0228	.2050	.0161	.2761	.0217	.2871	.0214
μ	.0132	.0009	.0131	.0013	.0178	.0011	.0179	.0013
E_0	737.0	62.8	661.2	31.3	620.6	70.5	409.2	18.7

Table 12: $J(u) \cdot 1000$ for the different weights in Simulation 2. The column represents the weight that is used for $J(u)$ in the Metropolis algorithm and the row shows the value of $J(u)$ for all four ω .

w.r.t. ω	weight ω			
	0	10^{-9}	10^{-8}	10^{-7}
0	8.7	8.2	9.2	9.6
10^{-9}	9.6	9.0	9.7	9.9
10^{-8}	17.2	15.7	14.7	12.8
10^{-7}	93.8	82.9	64.8	42.1

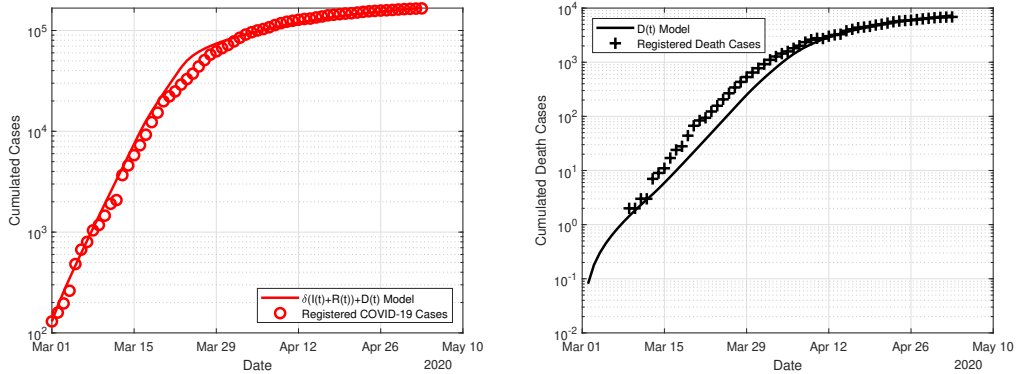


Figure 19: Plots for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 = 3$ and $\omega = 0$.

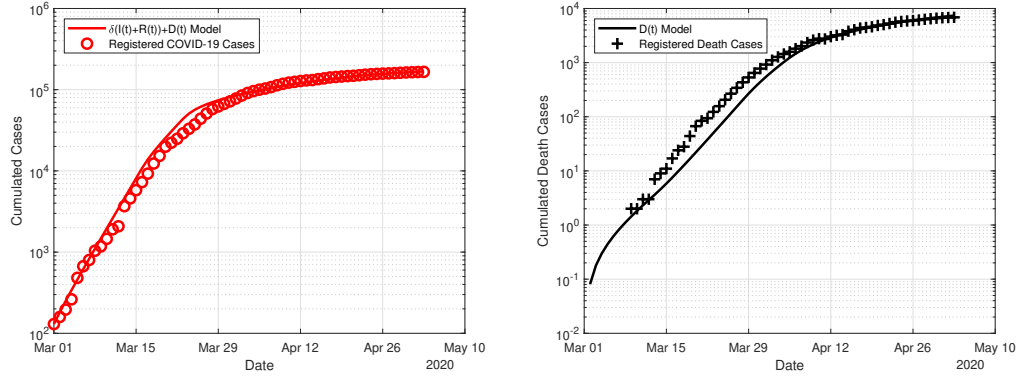


Figure 20: Plots for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-9}$.

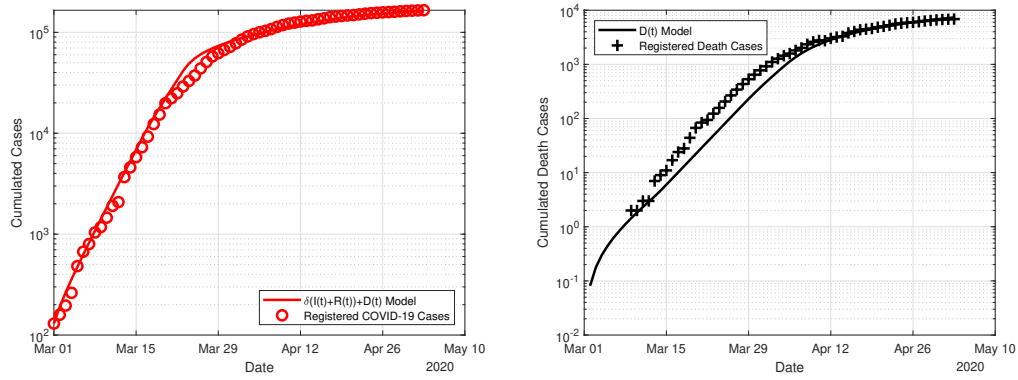


Figure 21: Plots for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 := 3$ and $\omega := 10^{-8}$.

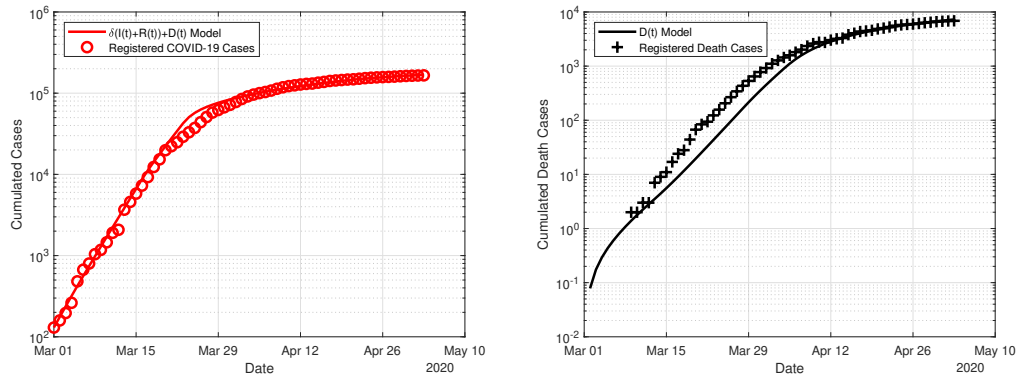


Figure 22: Plots for $\tau = 11.5$, $E_0 = \text{free}$, $I_0 = 114/\delta$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-7}$.

Simulation 3 – Free delay and initial infectives

Table 13: Estimates for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $R_0 = 16/\delta$, $\mathcal{R}_0 = 3$ after $r = 20000$ draws and using a step size of $s = u_0/100$.

Algorithm	$\omega = 0$		$\omega = 10^{-9}$		$\omega = 10^{-8}$		$\omega = 10^{-7}$	
	mean	std.	mean	std.	mean	std.	mean	std.
β_1	.5859	.0530	.6442	.0357	.6737	.0300	.7370	.0548
β_2	.4785	.0359	.6403	.0250	.5197	.0396	.4587	.0183
β_3	.0926	.0097	.0862	.0039	.0920	.0037	.0949	.0034
β_4	.0556	.0025	.0554	.0038	.0502	.0019	.0576	.0025
δ	.2768	.0295	.1911	.0115	.2063	.0135	.2237	.0155
μ	.0154	.0008	.0107	.0006	.0117	.0006	.0128	.0005
E_0	790.0	46.7	690.0	52.5	500.8	206.4	351.2	14.9
I_0	493.1	40.1	316.1	30.2	439.0	140.7	350.7	115.7
τ	7.3	.6	7.3	.4	7.4	.3	7.2	.6

Table 14: $J(u) \cdot 1000$ for the different weights in Simulation 3. The column represents the weight that is used for $J(u)$ in the Metropolis algorithm and the row shows the value of $J(u)$ for all four ω .

w.r.t. ω	weight ω			
	0	10^{-9}	10^{-8}	10^{-7}
0	3.8	3.3	3.4	4.1
10^{-9}	4.7	3.8	3.8	4.3
10^{-8}	12.5	9.0	7.8	6.5
10^{-7}	90.5	60.9	47.7	28.7

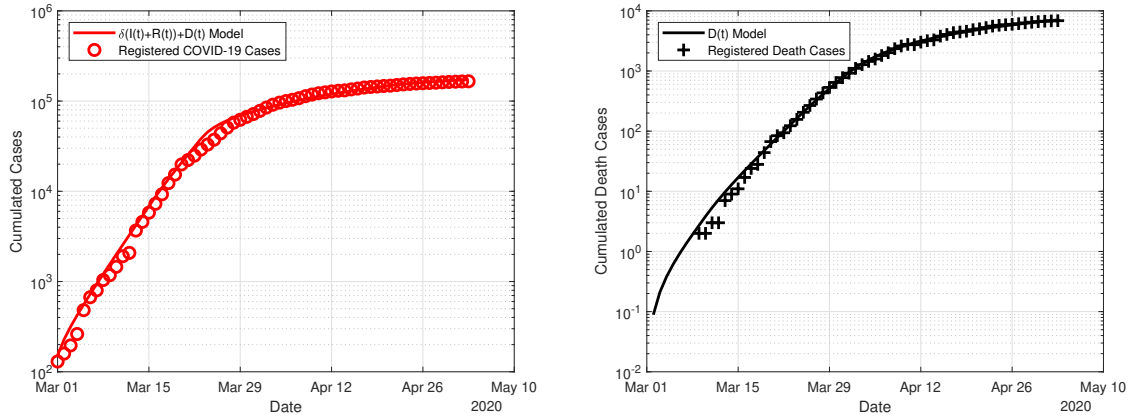
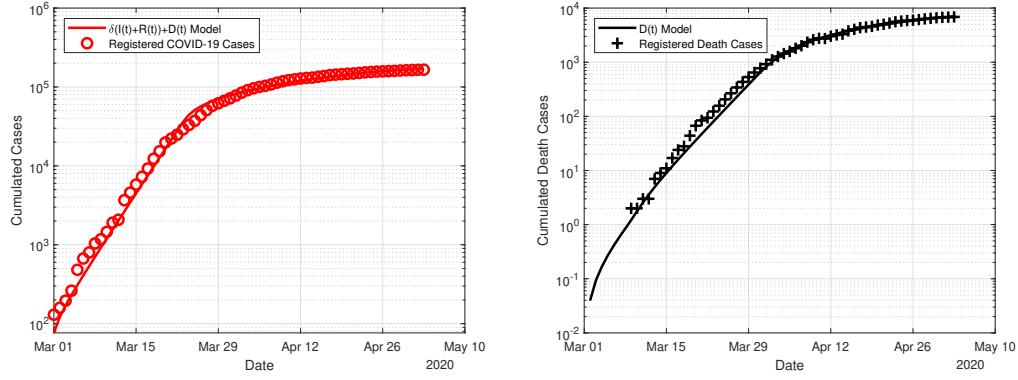
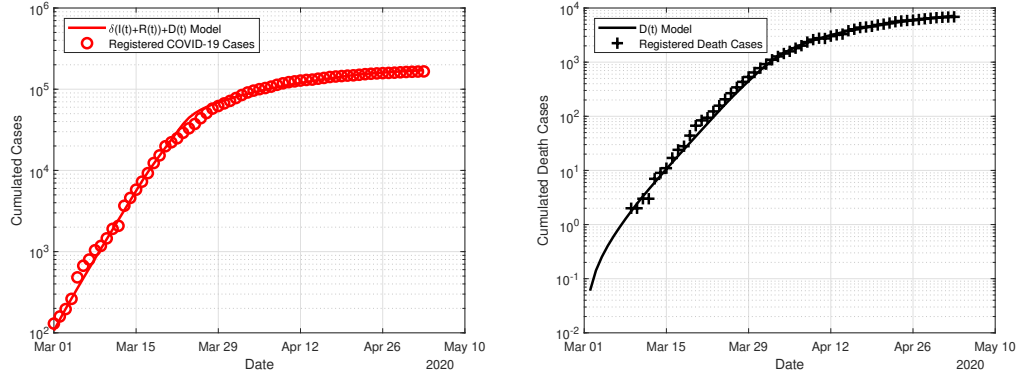
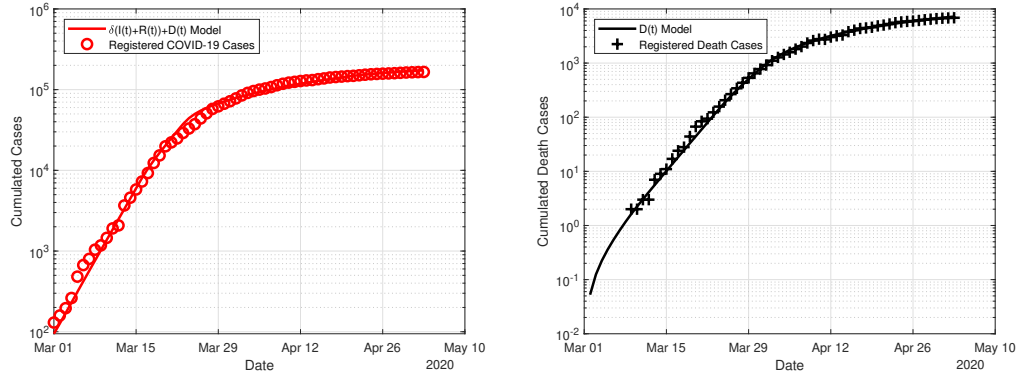


Figure 23: Plots for $\tau := \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega := 0$

Figure 24: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega := 10^{-9}$.Figure 25: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega := 10^{-8}$.Figure 26: Plots for $\tau = \text{free}$, $E_0 = \text{free}$, $I_0 = \text{free}$, $\mathcal{R}_0 = 3$ and $\omega = 10^{-7}$.