An Object-Oriented Solver for Modeling the Multi-regional COVID-19 Outbreak

Yujia Hao[1], Yuting Hou[1], Siwei Xu[1], Kai Chang[1], Zhen Wu[1], A. Veneziani[1,2]

[1]*Dept. of Mathematics, Emory University*
400 Dowman Dr NE, Atlanta (GA) USA
judy.hao@emory.edu, yuting.hou@emory.edu, siwei.xu@emory.edu, kai.chang@emory.edu, zhen.wu2@emory.edu
[2]*Dept. of Computer Science, Emory University*
400 Dowman Dr NE, Atlanta (GA) USA
avenez2@emory.edu

This paper presents the result of the work of a group of Undergraduate Students at Emory University, Atlanta (GA), USA (under the mentorship of A. Veneziani). We present a solver of a classical epidemiological model addressing the space-dependence of the outbreak by splitting a domain of interest (the "Domain") into subdomains (the "Geographical Units") connected by matrices representing the inter-regional mobility. We address the implementation of the solver with an Object-Oriented approach to promptly manage an arbitrary number of GU in the domain and to enable an easy, modular refinement of the local model.

Keywords: SIR-like models, Space-dependent models, Object-oriented programming

1 Introduction

The outbreak of a virus like COVID-19 depends on many factors at different scales. In a highly connected world, the disease rapidly spreads over different regions and evolve depending on local factors (including the different policies adopted by the different Governments). As such, the problem can be described by partial differential equations including both the time and space dependence of the different population compartments representing the different stages of the disease (Susceptible, Infected, Recovered, etc.). However, "traditional" space differential operators like the Laplacian can successfully describe short-range dynamics [1], while non-local mobility (e.g., flight connections) requires different approaches [2, 3]. In this paper, we present a space-dependent model where the space dependence follows the political organization of a Domain in Geographical Units - GU (e.g., the different States of the USA). Each Region is represented by a classical epidemiological compartment model, while the inter-region mobility is modelled by suitable matrices. The focus of the paper is on the implementation. To preserve modularity that could guarantee an easy management of the model in each Unit as well scalability on the number of Units, we opted for an Object-Oriented approach. The core is represented by the class of each GU; a Domain will be a class formed by a collection of Units with connecting matrices. We recall the basics of our GU model in Sect. 2, while inter-regional connections are described in Sect. 3. Details on the design of the solver are given in Sect. 4. Results are presented and validated in Sect. 5 on synthetic data. Limitations and conclusions are drawn in Sect. 6.

2 Mono-region SIR-Like Models for the Outbreak

In a classical SIR model for a specific region, the total population (with $N(t)$ individuals function of time t) is divided into three compartments: Susceptible $S(t)$, Infected $I(t)$, and Recovered $R(t)$. Similarly to a predator-prey model, the infection occurs proportionally to the product SI. SIR models are categorized into two models: the epidemic model that describes a rapid outbreak of the infectious disease and the endemic model that portrays the long-term dynamics by including regional birth and death rates (see, e.g., [4]). More precisely, the GU model

considered here reads

$$\begin{cases} d_t S = \mu N - \beta IS/N - (\mu + \rho)S \\ d_t I = \beta IS/N - \gamma I - \mu I \\ d_t R = \rho S + \gamma I - \mu R \end{cases} \tag{1}$$

Here, β is the number of the average adequate contacts for disease transmission per person per unit of time, and γ is the rate of recovery; μ is the reproduction/death rate assumed to be constant for simplicity. Should we set $\mu = 0$, dropping the reproduction/death rate (as reasonable in a short time range), we obtain the so-called "*epidemic*" model. The inclusion of vaccination is essential to its application in the real-world scenario. We assume that the vaccination rate is proportional to S with the factor ρ [5]. We are considering an idealized case, where vaccinated individuals are perfectly protected. More complex scenarios will be considered elsewhere. The vaccination coefficient ρ can be time-dependent as multiple factors can affect the vaccine distribution. We envision that the vaccination rate may range from a minimal value ρ_m, and increases to a maximum ρ_M as the organization and the advertising reach out for more people. We model this aspect by a logistic dynamics,

$$\rho(t) = \frac{\rho_m e^{a(t-t_V)}}{1 + \rho_m/\rho_M(e^{a(t-t_V)} - 1)} \tag{2}$$

where t_V is the first day of vaccinations. Notice that in this locqal GU model, the total number of individuals N is constant (summing up the three equations, for $N = S + I + R$ we obtain $d_t N = 0$. This is not true for multi-regional models, due to population mobility.

For the sake of notation, we set $\mathbf{A} \equiv [S, I, R]^T$ and the shorthand notation for the GU model (1) reads: $d_t \mathbf{A} = \mathbf{F}(\mathbf{A})$ where we omit for brevity the dependence of $\mathbf{F}$ on the parameters μ, β, γ, ρ.

3 Inter-regional Mobility

Let us consider now a set of n areas or GU denoted by the compartment vectors $\mathbf{A}_i$, $i = 1, 2, \ldots n$. Each area will have an intrinsic dynamic corresponding to the vector $\mathbf{F}_i(\mathbf{A}_i)$ as specified in the previous Section. Then, we consider the mobility among the regions as follows. As a simplifying assumption, we assume that people of a compartment in an area do not change compartment when moving: a susceptible in the GU j will keep the status of susceptible also in the target region i. The rate of susceptible moving from j to i will be $\sigma_{ij} S_j$ where σ_{ij} are mobility coefficients. In this way, the incoming rate of susceptibles in region i is given by $\sum_{j=1, j\neq i}^{n} \sigma_{ij} S_j$. Correspondingly, the outgoing rate of susceptible from the region i is $(-\sum_{j=1, j\neq i}^{n} \sigma_{ji}) S_i$. We can introduce the S-mobility matrix

$$M^S \equiv [m^S_{i,j}] = \begin{cases} \sigma_{ij} & \text{for } i \neq j \\ -\sum\limits_{k=1, k\neq i}^{n} \sigma_{ki} & \text{for } i = j \end{cases} \tag{3}$$

Similarly, we introduce an I-mobility and an R-mobility matrices M^I and M^R, and finally the block-diagonal $(n_c \times n)^2$ matrix $M = \mathrm{diag}(M^S, M^I, M^R)$, where n_c is the number of compartments per region (3 in our case). This matrix is promptly reordered consistently with the vector $\mathbf{A}$, oriented by geographical areas as opposed to a compartment-wise ordering. We keep calling M the rearranged matrix. In this way, the Domain, i.e. the set of GU, denoted by the vector $\mathbf{C} \equiv [\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n]$ will obey the ordinary differential system

$$d_t \mathbf{C} = \mathbf{D}(\mathbf{C}) + M\mathbf{C} \tag{4}$$

where $\mathbf{D}(\mathbf{C}) \equiv [\mathbf{F}_1(\mathbf{A}_1), \ldots, \mathbf{F}_n(\mathbf{A}_n)]^T$. Should we admit that a population can change compartment while moving across regions, the matrix M would have a full pattern (not block-diagonal).

As analytical solutions for this problem are not available, we opted for a numerical solver. In particular, we choose a second-order explicit solver like the Heun method, so to avoid the solution of non-linear equations.

Specifically, let us denote by Δt a user-selected time-step, and by t_k the collocation instants when we compute the solution. The Heun scheme then reads: given the solution $\mathbf{C}^k$ at time t_k, compute $\mathbf{C}^{k+1}$ as:

$$\begin{aligned} &\text{Predictor}: \quad \mathbf{C}^* = \mathbf{C}^k + \Delta t \left(\mathbf{D}(\mathbf{C}^k) + M\mathbf{C}^k\right) \\ &\text{Corrector}: \quad \mathbf{C}^{k+1} = \mathbf{C}^k + \frac{\Delta t}{2}\left(\mathbf{D}(\mathbf{C}^k) + \mathbf{D}(\mathbf{C}^*) + M(\mathbf{C}^k + \mathbf{C}^*)\right). \end{aligned} \tag{5}$$

At time t_0 the initial condition $\mathbf{C}^0$ is provided. The scheme is only conditionally stable: we empirically select our time-step to ensure numerical stability.

4 Design of the Object-Oriented Solver

One of the design specifics of our solver is to be flexible to both *horizontal* and *vertical* extensions. With this, we mean that we want a solver easily scalable to any given number of GU, so to manage the 20 regions in Italy (one of the Countries most affected by the virus [2]), the 50 States of the United States or even the counties [3] in one or more of the US (horizontal expansion). Meanwhile, we want to manage possible refinements of the models for the single area; while the SIR model is a good starting point, more compartments may be needed for a better representation of the outbreak (e.g. SEIR models); in an even more sophisticated modeling, we can consider Partial Differential Equations models, where the local mobility is assimilated to a Brownian motion [1] (*vertical flexibility*). For this reason, *Object-Oriented* programming provides the right framework to our solver, particularly for the features of encapsulation and inheritance.

The GU Class (local model) The core class of our solver is the GU Class implementing the model for the single area represented by the generic equation

$$d_t\mathbf{A} = \mathbf{F}(\mathbf{A})$$

with the corresponding initial conditions. At the basic level, the class implements the SIR model in (1). As members of the class we include:

- the vector $\mathbf{A}$ (by default given by S, I, R, with the number of compartments $n_c(= 3)$);
- the vector $\mathbf{F}$ with the associated parameters β, μ, γ, ρ, possibly function of time;
- the initial conditions.

More sophisticated models are inherited by this class. For instance, to implement a PDE-based model like [1] we can add as members the mesh of the region of interest. Similarly, to include the structure of age, we can extend the number of compartments and parameters to describe Susceptible etc. at the different ages, by exploiting the inheritance from the basic class.

The Domain Class (global model) To cover the entire domain of interest, we introduce a class with the following members:

- an array of GU;
- the mobility matrix M;
- a solver for the problem (4) with the initial conditions of the GU, for instance (5).

The solver was implemented both in Python and Matlab (in the latter case, the classes are given by appropriate structs) and it is available under GPL at the GitHub repository `github.com/HowardYutingHou/CovidProject`.

5 Benchmarks and Results

We present a number of tests we run to check the correctness of our solver. Specifically, we tested the accuracy of our simulations against analytical results, when available. Then, we used our two different and independent implementations to cross-validate the results on a fictitious benchmark consisting of seven regions (having the nickname of seven States of the U.S). Successively, we tested the endemic vs the epidemic models, to stress the importance of the parameter μ in the final results. In the second part of this Section, we illustrate the results we obtain when we want to mimic the presence of lockdown policies and a vaccination campaign. In this paper, we consider only basic SIR models. A more complete presentation of the results will be provided in a forthcoming paper.

5.1 Consistency test with theory for one-region models

When cross-regional interactions are disregarded, each local dynamics is represented locally by the classical epidemic or endemic SIR models. In this case, analytical results are available [4], Theorems 2.1 and 2.2. For example, setting $\vartheta \equiv \beta/(\gamma+\mu)$, Theorem 2.2 states that if $\vartheta \leq 1$, the solutions approach disease-free equilibrium; if $\vartheta > 1$, the solutions should reach the endemic equilibrium $S_e = N/\vartheta$, $I_e = \mu(\vartheta - 1)N/\beta$. The consistency of our simulations with these results was carefully assessed and verified.

5.2 Cross-validation

Another test was performed by comparing the results of the two different codes developed by the team, with two different languages (Matlab and Python). To this aim, we introduced a fictitious data-set, with seven GU. For the sake of notation, we gave these units the name of seven States of the US. The parameters and the mobility matrix were however selected without a specific tuning of the data, so the names are to be intended just as nicknames, with no particular reference to the "real" States. Fine tuning of the parameters is, in fact, an important follow-up of the present work. For the present proof-of-concept, we calibrated the results arbitrarily, based on the available literature. Figure 1 shows the computational solutions of the GUs "Georgia" and "Alabama" with the endemic model from both versions of the solver. By visual and quantitative inspections, the results do coincide. The two

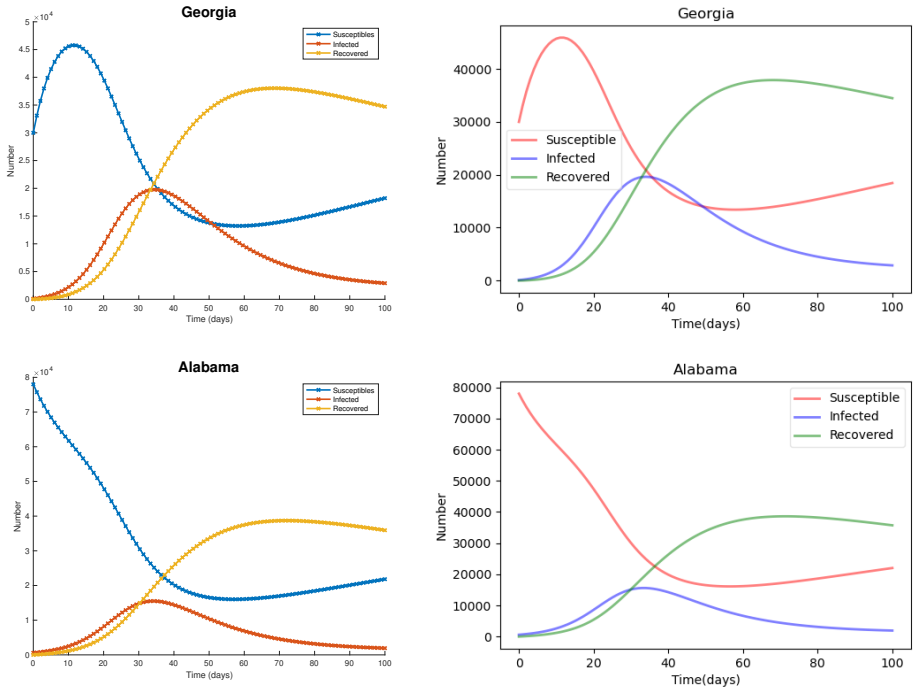

Figure 1. Left: Results for our multi-regional MATLAB solver in "Georgia" and "Alabama"; right: results for the same case with the Python solver.

solvers were developed independently by two subgroups of the authors, with just the same code-specifications. The comparison provides therefore an excellent cross-validation of our results, giving confidence in the correctness of the solvers.

5.3 Epidemic vs Endemic

Figure 2 shows the comparison of the multi-regional solver in "Georgia", when we drop the local reproduction rate ($\mu = 0$), i.e. the "epidemic model", on the left, and when we include it ("endemic model") on the right. The increment of S in the first 10 days is due to the mobility (particularly, in this case, from "California" and "Texas", with a mobility rate of 0.01 and 0.02 respectively). The comparison between the two plots highlights the importance of the coefficient μ on the long time range, as the endemic model exhibits and additional increment of S, absent in the endemic one.

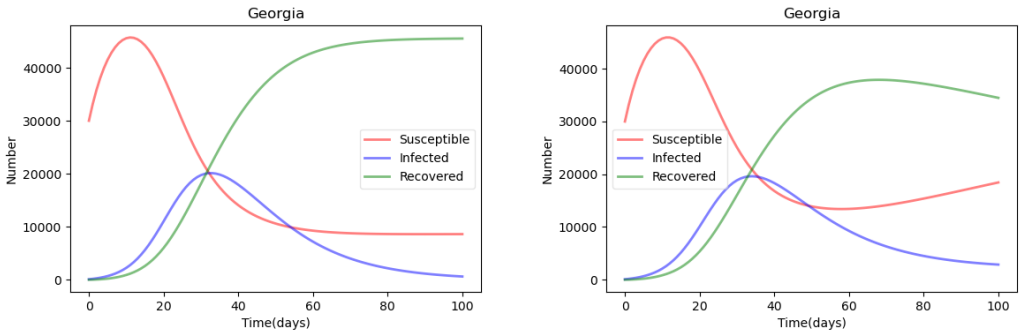

Figure 2. Results in "Georgia": left: epidemic model ($\mu = 0$); right: endemic model.

5.4 Effects of Lockdown

We simulate the effects of lockdown restrictions by reducing the infection rate β of a factor of 10 as a result of restrictive measures like mask-enforcement, curfew, etc.

Local Lockdown We consider first the effects of lockdown in an isolated region or GU. In Fig. 3 we show the results in "Goergia" with no mobility: on the left the no-lockdown results for 200 days, on the right the results with a lockdown enforced from day 20 to 60. The impact of the lockdown is evident, particularly on the I population. At the end of the restrictive measures, there is a partial comeback, however the lockdown significantly helps containing the number of infected (from 10000 to 5000). At the end of the lockdown there is a spike of infected, as the number of susceptible is considerably higher than in the no-lockdown case, yet the peak of I is halved.

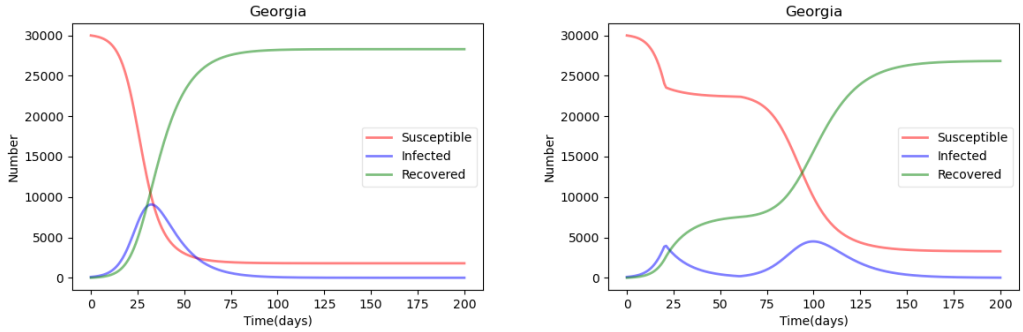

Figure 3. Local evolution in "Georgia" without lockdown (left) and with lockdown (right).

Multi-regional Lockdown

We consider now the multi-regional case: during lockdown, β is reduced by a factor of 10 and the borders are closed, i.e. the mobility coefficients are set to 0. In particular, we assume that in the 7-state case, only Georgia enforces a lockdown (Fig. 4). As expected, with the restrictions on the mobility, the results are similar to the isolated case, and the lockdown reduces the number of infected to about 10000 vs the 20000 of the no-lockdown case.

We also plot in Fig. 5 Florida and New York from day 0 to day 100 to show how a lockdown in Georgia can slightly affect other regions. After day 60, Florida immediately experiences an increase in S because once the lockdown is dismissed, 0.03 of Georgia's susceptible (incremented by the lockdown) move into Florida (only 0.02 of Florida's S is assumed to move into Georgia). As for New York, there is a slower increase in R after day 60. As 0.01 of Georgia's recovered population is moving into New York, and the recovered population of Georgia is lower than without lockdown, while 0.015 percent of New York recovered is moving into Georgia. In conclusion, if the lockdown is enforced only on one State, the effects for the other States appear to be minor.

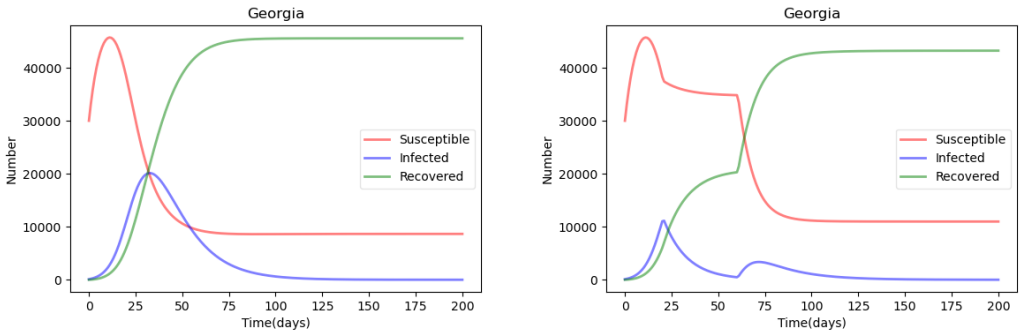

Figure 4. left: without lockdown; right: with lockdown.

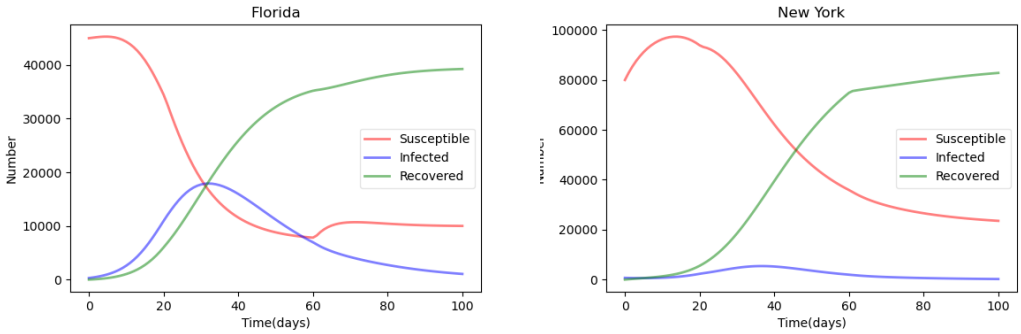

Figure 5. Multi-regional case with a lockdown in Georgia: left: Florida; right: New York.

5.5 Effects of Vaccination

To test the effects of vaccination (in absence of lockdown), we introduce the vaccine at day 80, where the vaccination rate is modeled by (2). We refer again to "Georgia". We observe that at day 80, the development pattern of the recovered is interrupted and R increases at a much higher rate. At the same time, the population of the susceptible drop significantly, decreasing faster than before.

Notice that the total number of recovered reaches a greater amount than that without vaccination. Also, the susceptible population decreases to 0 at the end, which did not happen where vaccination was not implemented, reaching the herd immunity. Therefore, our model suggests that the introduction of vaccination can end the outbreak. We compare the effects of a temporary measure like the lockdown and the vaccination in Fig. 6.

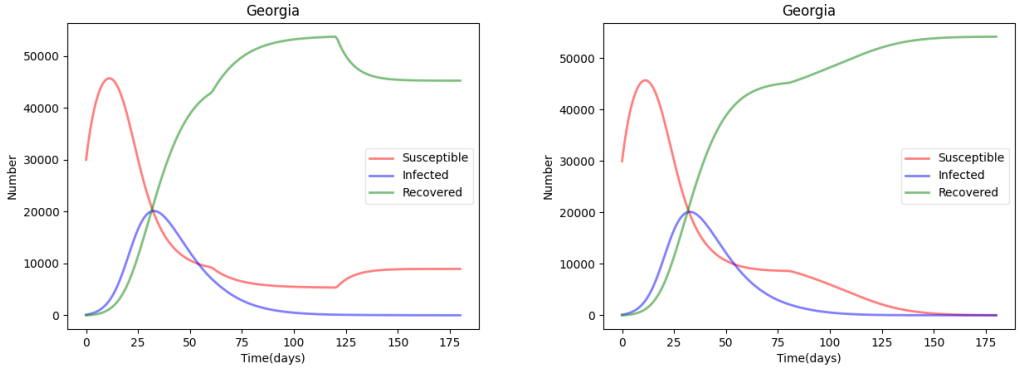

Figure 6. Measurements for fighting the pandemic in a multi-regional solver, with focus on "Georgia": left, lockdown, right, vaccination.

6 Conclusions: Limitations and Perspectives

This work represents a proof-of-concept for a tool developed by undergraduate students to include the space-dependence in the pandemic based on a geographical (or political) subdivision in Units connected by mobility data. We validated our solver against the theory and by cross validation of two solvers developed independently with different languages. This is just a starting point for a tool to test in realistic and complex scenarios (e.g. [2]). Here a list of improvements we plan to add to the current solver.

Local Model The SIR model is just a starting point. Class inheritance enables using more complex models like SEIR [2], up to models incorporating a local space dependence like in [1]. In the latter case, we will combine a multi-scale representation of population mobility, with a PDE for local range and matrices for the long range. Inheritance of classes should guarantee a prompt extension of the current solver when all the GU are modeled by the same system, more challenging will be the implementation of heterogeneous local solvers (e.g. with complex models in some regions and simpler in others). Other improvements refer to the introduction of a structure of age (particularly evident in the COVID-19 pandemic); the introduction of re-infection of vaccinated people and virus variants in the model.

Numerical Methods The Heun method is explicit, so to avoid the computational burden of nonlinear solvers, consequently it is only conditionally stable. This requires a manual tuning of the time-step, to avoid numerical oscillations. We are considering the introduction of adaptive explicit solvers (like the Runge-Kutta Fehlberg 45), where the time step is selected automatically to guarantee the accuracy. Notice that the region of stability of the multi-regional solver is driven by the smallest stability region of the single GU. A challenging task will be the introduction of a solver-by-subdomains, where the solution of the global domain is achieved by solving iteratively the subdomains with different time-steps selected adaptively. This may be crucial in a multi-regional multi-scale solvers based on PDEs to contain the computational costs.

Parameter Tuning The local parameters and the mobility data need to be retrieved from measurements. In our results we just performed an empirical tuning. Clearly, the reliability of the results strongly depend on those parameters. We argue that, even in presence of complete measures, the parameters will be affected by noise. Beyond the data assimilation required by fine-tuning (the measures refer to the populations and not to the parameters directly) [6], we will consider a specific sensitivity analysis to assess the impact of the noise on the results for the different parameters. Either minimal variance techniques for non-linear problems like the Ensemble Kalman-Filter or Bayesan approaches [7] can be considered. In our case, we plan specifically to pursue Deep Learning approaches inspired by [3].

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] A. Viguerie, G. Lorenzo, F. Auricchio, D. Baroli, T. J. Hughes, A. Patton, A. Reali, T. E. Yankeelov, and A. Veneziani. Simulating the spread of covid-19 via a spatially-resolved susceptible–exposed–infected–recovered–deceased (seird) model with heterogeneous diffusion. *Applied Mathematics Letters*, vol. 111, pp. 106617, 2021.
[2] M. Gatto, E. Bertuzzo, L. Mari, S. Miccoli, L. Carraro, R. Casagrandi, and A. Rinaldo. Spread and dynamics of the covid-19 epidemic in italy: Effects of emergency containment measures. *Proceedings of the National Academy of Sciences*, vol. 117, n. 19, pp. 10484–10491, 2020.
[3] M. A. Bhouri, F. S. Costabal, H. Wang, K. Linka, M. Peirlinck, E. Kuhl, and P. Perdikaris. Covid-19 dynamics across the us: A deep learning study of human mobility and social behavior. *Computer Methods in Applied Mechanics and Engineering*, vol. 382, pp. 113891, 2021.
[4] H. W. Hethcote. The mathematics of infectious diseases. *SIAM Review*, vol. 42, n. 4, pp. 599–653, 2000.
[5] B. D. MacCluer, P. S. Bourdon, and T. L. Kriete. *Differential Equations: Techniques, Theory, and Applications*. American Mathematical Soc., 2019.
[6] M. Asch, M. Bocquet, and M. Nodet. *Data assimilation: methods, algorithms, and applications*. SIAM, 2016.
[7] D. Calvetti, A. Hoover, J. Rose, and E. Somersalo. Bayesian dynamical estimation of the parameters of an se (a) ir covid-19 spread model. *arXiv preprint arXiv:2005.04365*, vol. , 2020.