PRE-PRINT MANUSCRIPT

SimCOVID: An Open-Source Simulation Program for the COVID-19 Outbreak

Ismael Abdulrahman, ismael.abdulrahman@epu.edu.iq
Information System Engineering Department, Erbil Technical Engineering College, Erbil Polytechnic University, Iraq

Abstract—This paper presents an open-source computer simulation program developed for simulating, tracking and forecasting the COVID-19 outbreak. The program is built in Simulink with a block diagram display. The mathematical model used in this program is the SIR and SEIRD models represented by a set of differential-algebraic equations. It can be easily modified to develop new models for the problem. The program uses the outbreaks of China and Italy as test models. The infection and recovery rate functions are treated as constant, variable, or a combination of the two. In addition, an adaptive neuro-fuzzy inference system is employed and proposed to train the model and predict its output. The program showed good matching between the simulated and the reported cases and it predicts the final size of the Italy outbreak to be in the range 230,000–330,000 cases as of July 2020.

Index Terms— Coronavirus, COVID-19, Outbreak, Simulation Program, Simulink.

1. INTRODUCTION

Coronavirus disease 2019 (known as COVID-19 or 2019-nCoV) is a disease caused by a novel virus called SARS-CoV-2 [1]. The first reported case of this disease was on Dec. 31, 2019, in Wuhan, China. The outbreak has been declared as a public health emergency of international concern by the World Health Organization (WHO) on Jan. 30, 2020 [2], and as a pandemic on March 11 [3]. The virus spread rapidly around the world and several large-size clusters of the spread have been observed worldwide including outbreaks in China, Iran, Italy, Spain, France, UK, and the USA [4]. According to Johns Hopkins University, the total number of confirmed cases surpassed one million cases on April 2, 2020 [5].

An essential part of minimizing the spread of the virus is to monitor, track, and estimate the final size of the pandemic. This is extremely useful for decision making against the public health crises [6]. One way to predict the dynamic spread of the epidemic is by the use of computer simulation following the mathematical model of an epidemic. In the literature, several analytical approaches have been proposed to model the pandemic including Susceptible-Infected-Removed (SIR) model [6-7], Susceptible-Exposed-Infected-Removed (SEIR) model [1], Susceptible-Infected-Recovered-Dead (SIRD) model [8,9], and fractional-derivative SEIR [10], and SEIRD [11]. While some recent studies are addressing this epidemic using the aforementioned models [6-11], there is an increasing need to develop an open-source computer program to perform a time-domain simulation of the dynamic spread of the virus. References [12-13] present MATLAB code scripts to achieve this objective. Reference [14] presents a Python-based program called CHIME (COVID-19 Hospital Impact Model for Epidemics) for hospital uses.

It is advantageous to have an educational program that displays the mathematical model of the epidemic in a visualized block diagram instead of a coded script. One of the widely-used platforms for studying the dynamic behavior of a system is Simulink. It has been used by many academic researchers in different fields for simulating a system using time-domain simulations. A dynamic system, such as the COVID-19 epidemic, can be mathematically modeled by a set of differential equations (DEs) or a set of differential-algebraic equations (DAEs) depending on the employed model. Simulink provides the user with useful mathematical tools including parameter estimation and system optimization which are needed in simulating a pandemic such as COVID-19. Simulink allows the user to enter and plot the confirmed cases of infection of the disease as an input to the system and then compute the system parameters (for instance, infection rate, recovery rate, etc.) so that the output of the simulation and the collected actual data are equals or very close to each other.

This paper presents an open-source program for tracking, estimating, and simulating the coronavirus outbreak. Unlike the existing models, the mathematical model of the outbreak is represented and visualized as a block diagram in Simulink using SIR and SEIRD representation. The infection and recovery rate functions are treated as a constant, variable, or a combination of them. With the application of the adaptive neuro-fuzzy inference system (ANFIS), the outbreak of China and Italy are implemented in Simulink using both a standard mathematical model and ANFIS system.

1.1 SIR, SEIR, and SIRD models

SIR model is a basic representation used widely to describe a disease spread, and it is the fundamental model for the other models such as SEIR and SIRD. SIR model consists of three-compartment levels: Susceptible, Infectious, and Removed. Any individual belongs to one of these groups. A brief description of these compartments is given below. Susceptible individuals are those people who have no immunity to the disease but they are not infectious. Since there is no vaccine yet developed for this disease, we can say that the entire community is exposed to get infected by this disease and hence, the "Susceptible" compartment can be represented by the entire population. An individual in the "Susceptible" level can move into the next level of the model (Infectious) through contact

Fig. 1a SIR model

Fig. 1b SEIR model

with an infectious person. By this single transmission, the number of susceptible\infectious people reduces\increases by one, respectively. The next group of people is for the infectious people who have the disease and can spread it to susceptible people. Infectious people can move to the "Removed" compartment by recovering from the disease. The removed compartment includes those who are no longer infectious and the ones who have dies from the disease (closed cases). The summation of these three compartments in the SIR model remains constant and equals the initial number of population. A basic SIR model is shown in Fig. 1, where β denotes the infection rate or the transmission rate or the force of infection, and γ denotes the recovery or removed rate. Generally speaking, these parameters (β, γ) are not constant; they are functions of the size of infectious and recovery compartments. These are the parameters that we want to optimize and estimate so that the reported and simulated cases are approximately equals. To solve this set of differential equations, we need initial values for the three-state variables S, I, and R namely $S_{(0)}$, $I_{(0)}$, and $R_{(0)}$. The initial value $S_{(0)}$ is the community population impacted by the outbreak [6], whereas, $I_{(0)}$ is the number of confirmed cases that can be any value but not zero. We can set $R_{(0)}$ to zero if the start times of the spread and simulation are equals. The transmission rate reduces monotonically with time [6].

Mathematically, a standard SIR model can be represented using the following differential equations:

$$\frac{dS}{dt} = -\frac{\beta}{N} SI \tag{1}$$

$$\frac{dI}{dt} = \frac{\beta}{N} SI - \gamma I \tag{2}$$

$$\frac{dR}{dt} = \gamma I \tag{3}$$

where N denotes the total population size, $N = S + I + R$.

For the SEIR model, an additional compartment is added between the Susceptible and Infectious compartment called "Exposed". This compartment is dedicated to those people who are infectious but they do not infect others for a period of time namely incubation or latent period. Note that the summation of the four state-variables at any time stays constant, and λ refers to the incubation period. For the SIRD model, an additional compartment is added at the end of the SIR model to distinguish between recovered and death cases in the Removed compartment. Note that the $\frac{\beta}{\gamma}$ ratio gives us an important metric called "basic reproductive number", or R_0. R_0 is a measure of how contagious a disease is, in other words, how many people are infected by a single infected person. If $R_0 > 1$, there is a spread of disease which is a strong sign of a pandemic. If $R_0 < 1$, there is a decline in the spread [15].

2. Simulation

The differential equations (1)-(3) are built in Simulink using the built-in blocks available in the library browser. Different colors are used to distinguish these equations (blue for "Susceptible", red for "Infectious", and green for "Removed" compartment). Note that the control measure parameters (β, γ) are functions of time to be found and estimated by the program. Figure 2 shows the block diagram of the model.

Since reliable long-time data is available only for the case of China, we will first consider this test case and simulate it using the SIR model. All the numerical data for the test cases were taken from [16]. The collected data (officially reported cases) displays some inconvenience observations in the reported cases of infection (the evidence is the jump in the curve on Feb. 13th) which is reportedly [6] caused by a sudden change in the way the data was collected.

The proposed method to estimate the variable parameters is to divide the curve (either the daily-infection-rate curve or the cumulative-infectious curve) into at least two parts around some observed critical points. The objective is to cover and better fit the concave-up and -down portions of the curve. In this study and for more reliable results, the cumulative reported cases graph is split into three periods at two critical points: (1) period-1 from time zero until $st1$, where st stands for starting time, (2) period-2 from $st1$ to $st2$ and (3) period-3 from $st2$ until the rest of simulation time. The critical time points ($st1$ and $st2$) were chosen empirically based on some observation notes about the given curve. It is also possible to let these values variables to be found by the program. The total period of time was chosen to be from the day when the first case of the disease was reported until the curve starts to flatten. Three step-blocks were used for the parameters (β, γ). These quantities are unknown variables; we want to determine their values and the overall shape forms so that the output of our simulation and the collected data are equals. The parameter estimation tool in Simulink was employed to optimize the system and predict the infection and recovery rates. A reasonable set of initial values were selected for the parameters (positive values between 0-2 for the recovery gains and negative similar values for the exponential powers used for the infection function).

Figure 3 shows a part of this simulation with the number of iterations, values of the parameters at each iteration, and the comparison between the collected and simulated data. We can observe a quite matching between the two curves after 100 iterations. It should be noted that the selection of initial values given to the optimization algorithm plays a significant role in the speed to reach the desired accuracy. Therefore, it is recommended to set reasonable values for these quantities to accelerate the simulation. An important indicator of a pandemic is the ratio between the infection and recovery rates, or namely, the reproduction ratio.

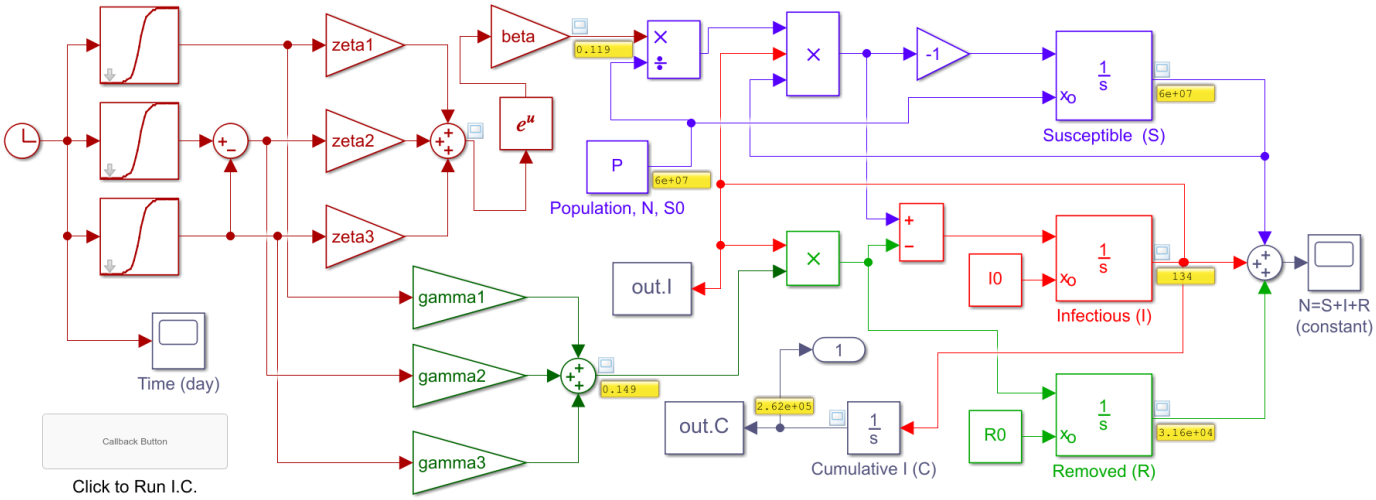

Fig. 2 Block diagram of the SIR model in Simulink – blue is for "Susceptible" with its input –beta function–, red is for "Infectious", and green is for "Recovered" with its input –gamma function

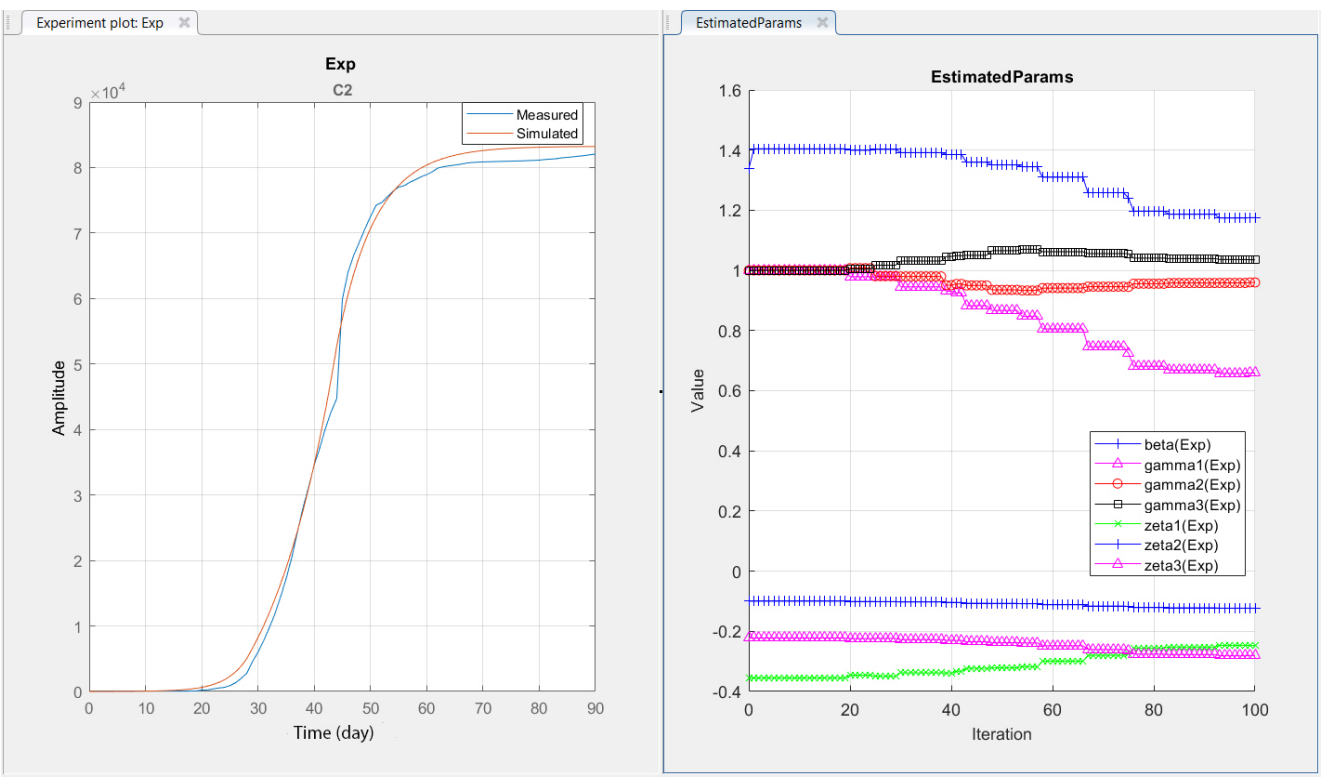

Fig. 3 Simulated and reported cumulative infectious plots (left) and the optimized parameters at each iteration (right)

Figure 4 shows the final values of the optimized parameters for the outbreak of China and Italy. We can see a reproductive ratio of more than one (a sign of pandemic) from the beginning of the outbreak until around the inflection point of the curve. The numerical results obtained from this simulation are plotted and shown in Fig. 5a for the China outbreak and Fig.5b–c for the Italy outbreak. The daily infectious state variable is plotted on the left y-axis whereas the cumulative infectious variable is plotted on the right y-axis. Note that the cumulative infectious is the sum of the daily infectious values which can be modeled in Simulink using an integrator. As mentioned in [6], the sudden jump in the curve is due to the way the data was collected and the diagnosis techniques which is not interpreted by a natural variation of a pandemic. Therefore, the program tries to change the parameters so that the simulated cumulative cases are close to the actual recorded ones. There is a good matching between the two cumulative plots although the daily infectious plots show some mismatch at peak time. For the Italy outbreak, there is a quite matching between the simulated and the reported cases as there is no change in the calculation method. In addition, we need to estimate the final size of the epidemic. The program estimates the rest of the simulation and the results show that the total cases for Italy outbreak could reach 230,000–330,000 cases at the end of July 2020 with a ± 10 error in the infection rate. This estimation is based on the given data from the beginning of the reported cases until the date April 09, 2020. For the China outbreak, it is enough to take data for simulation until the settling time of the curve which is around middle of March 2020. Fig. 5b–c show the daily infectious and cumulative infectious plots for the Italy outbreak with the estimated final range colored in grey [17].

It is worth mentioning that Simulink provides several optimization techniques that can be used for this problem including Gradient Descent, Nonlinear least Square, Pattern Search, and Simplex Search. In addition, various options are included in the toolbox. For instance, the Gradient Descent technique allows applying Active-Set, Interior-Point, Trust-Region-reflective, and\or Sequential Quadratic-Programming methods. Nonlinear least Square technique comes with either Levenberg-Marquardt or Trust-Region_reflective strategy. Pattern Search strategy involves Positive Basis NP1 and 2N, Genetic Algorithm, Latin Hypercube, and Nelder Mead techniques. Furthermore, the cost function can be selected as either Sum Squared or Absolute Error. From the simulation experience, it is found that the Simplex Search technique shows faster running time but it ignores the limits specified for the problem.

3. Detailed Model of SIR (SEIR and SIRD)

The SIR model described above does not give information on the exposed people who are infected but not detected yet (not confirmed yet). It does not provide any knowledge of the closed cases of infectious people who have passed away. An exposed variable can be added to the SIR model to form a SEIR model, whereas a closed (death) variable can be added to the SIR model to form a SIRD model. These two variables can also be added together to the SIR model and form a new model called SEIRD. This model is more general and is adopted in this section [11]. The differential equations of this model are as follows:

$$\frac{dS}{dt} = -\frac{\beta}{N} SI \tag{4}$$

$$\frac{dE}{dt} = \frac{\beta}{N} SI - \lambda E \tag{5}$$

$$\frac{dI}{dt} = \lambda E - \gamma I - \delta I \tag{6}$$

$$\frac{dR}{dt} = \gamma I \tag{7}$$

$$\frac{dD}{dt} = \delta I \tag{8}$$

Where E refers to the exposed state variable, D refers to the closed or death case, λ refers to the exposed parameter, and δ refers to the dead variable. Summing (4)–(8) should give zero whereas the total sum of the state variables should be constant and equals the population:

$$N = S + E + I + R + D \tag{9}$$

A new Simulink program was developed for this model and is shown in Fig. 6. Note that the actual collected data was used as input to this model. This data is stored in Excel sheets and there is a block in Simulink that allows us to import data from external sources and build a customized signal. The name of the block is “signal builder”. The ratio between the daily dead and daily infectious is used to represent the mortality rate. This signal represents an actual data-based signal which is multiplied by the simulated infectious signal to be integrated and form the cumulative dead variable using an integrator block in Simulink. In the figure, orange color is used for “Exposed” and dark red is used for “Dead” compartments. Using the same initial values used for the SIR model, the model gives similar results comparing to the standard model but with two new variables, the exposed and mortality cases. The parameter optimization process and the numerical outputs are not plotted here owing to its similarity to the above model and for space limitation reasons. However, all these models and programs are provided with this paper.

4. Simulation with the Effect of Varies Control Measures

Total or partial lockdown, social distancing, and stay-at-home control steps can influence the spread of the virus and flatten the curve earlier if these steps were applied in advance. In the Simulink model, we can represent all these control measures by a step function and simulate the system under these conditions. These controls affect the infection rate giving a decline in the beta function. On the other hand, developing a vaccine or involving any other ways to recover or slow down the disease (such as providing the hospitals with all the necessary ventilators) affect the recovery-rate parameter by increasing this value. As a result, the reproduction ratio reduces by any change in these two parameters. It is also easy in Simulink to see the impact of a delay in response on the curve. This can be achieved by adding a delay block to the infection function. However, more details about the model are needed for this problem.

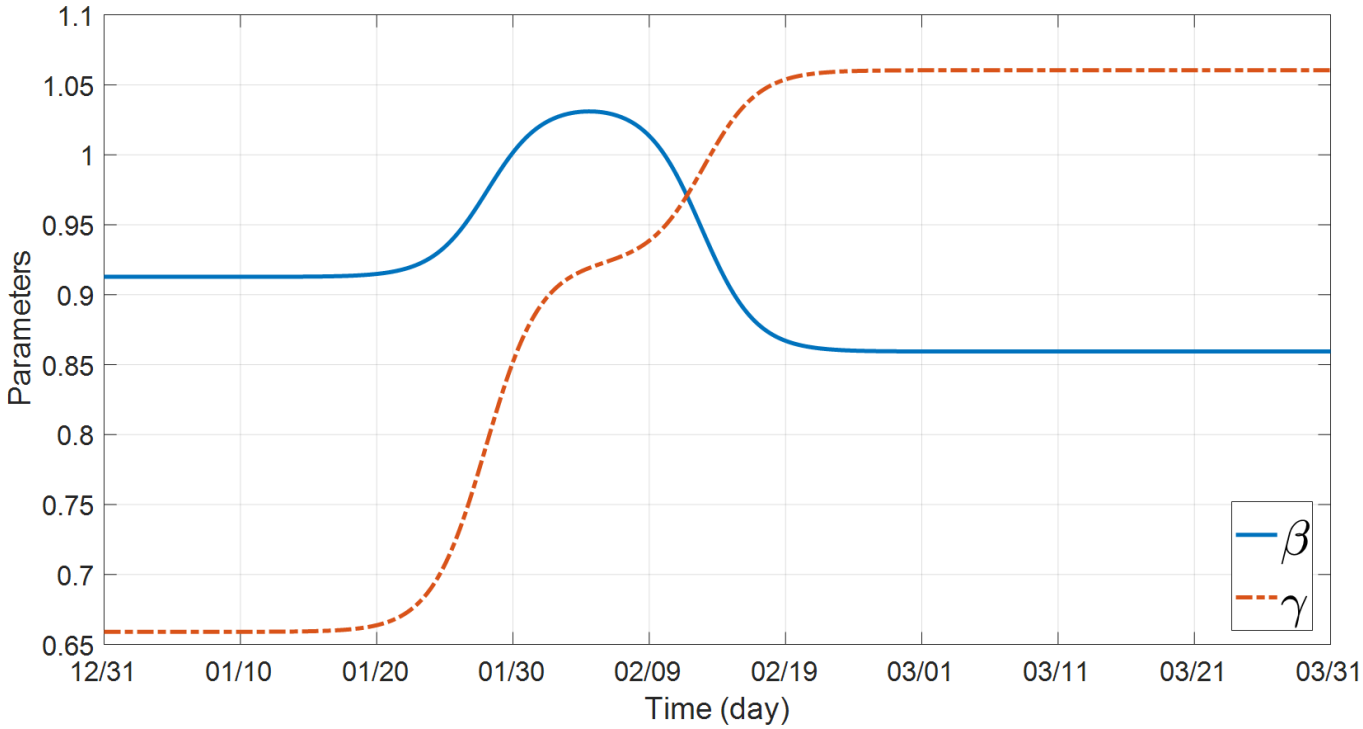

Fig. 4 Final values of the infection rate (solid blue) and recovery rate (dashed orange) parameters for the China outbreak

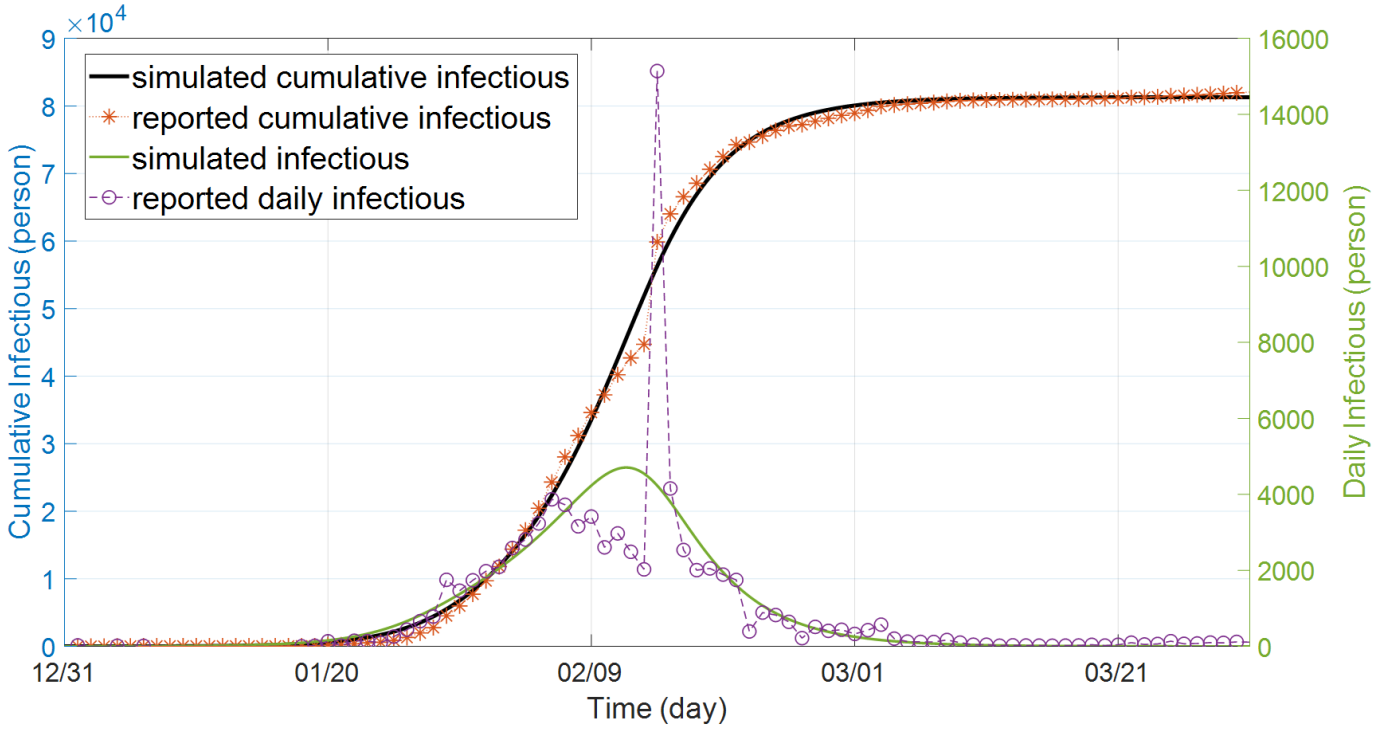

Fig. 5a Simulated and reported cases (daily infectious on the right y-axis and cumulative on the left y-axis) – case of China

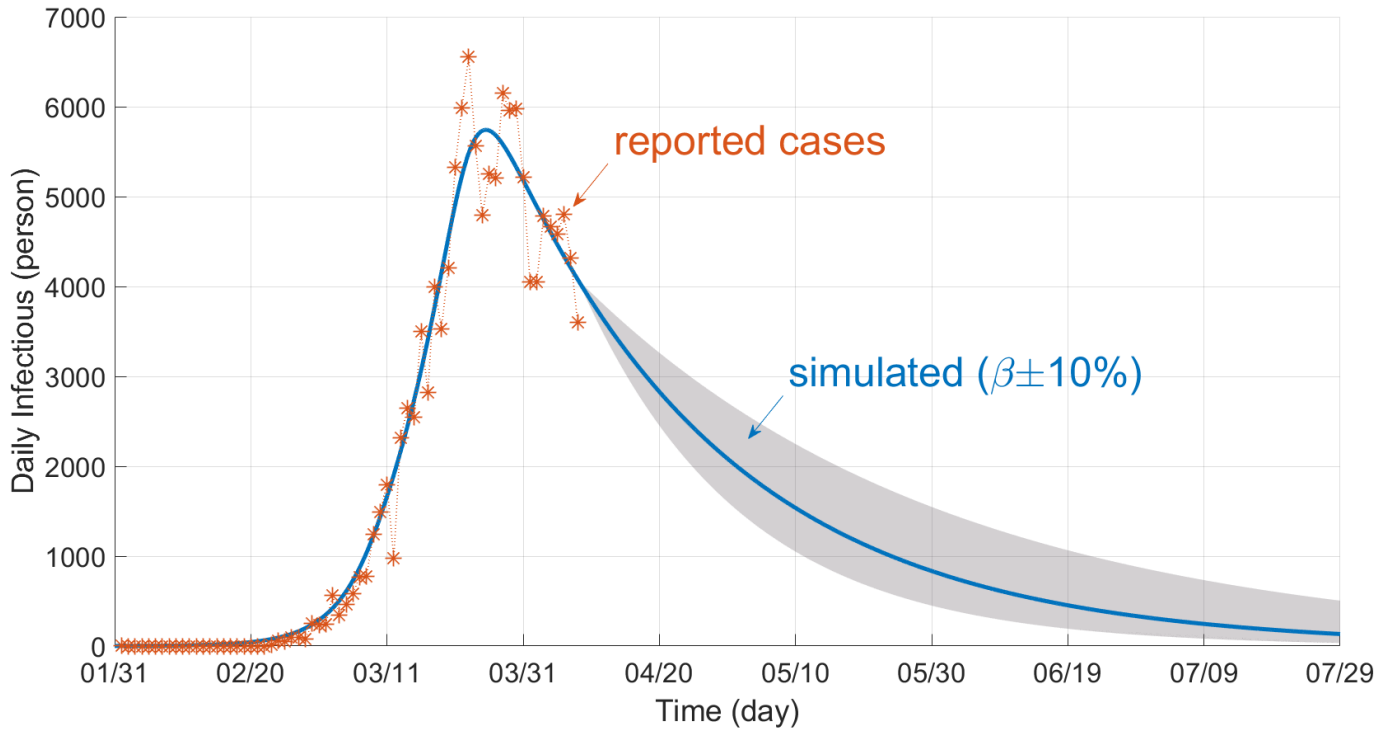

Fig. 5b Simulated and reported daily infectious – case of Italy–, with prediction of the final size of the pandemic

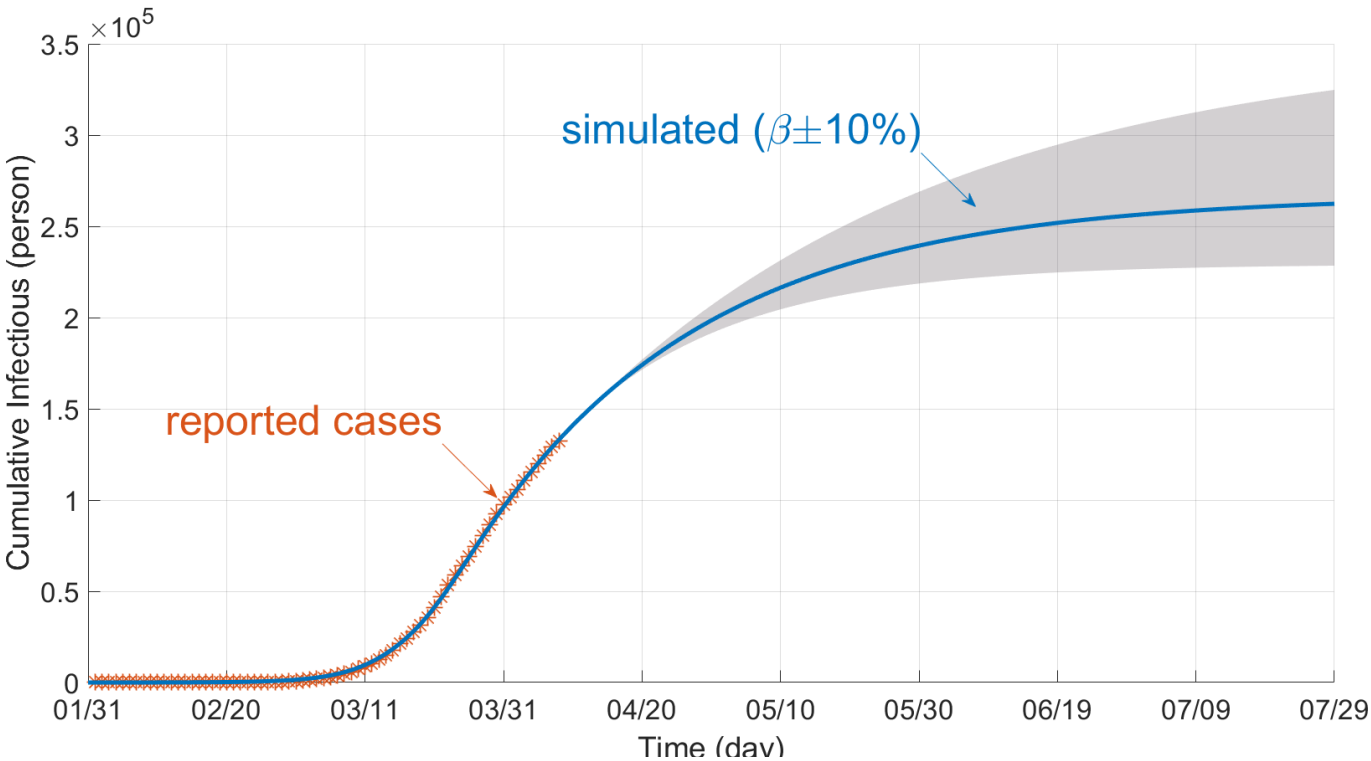

Fig. 5c Simulated and reported cumulative infectious – case of Italy–, with prediction of the final size of the pandemic

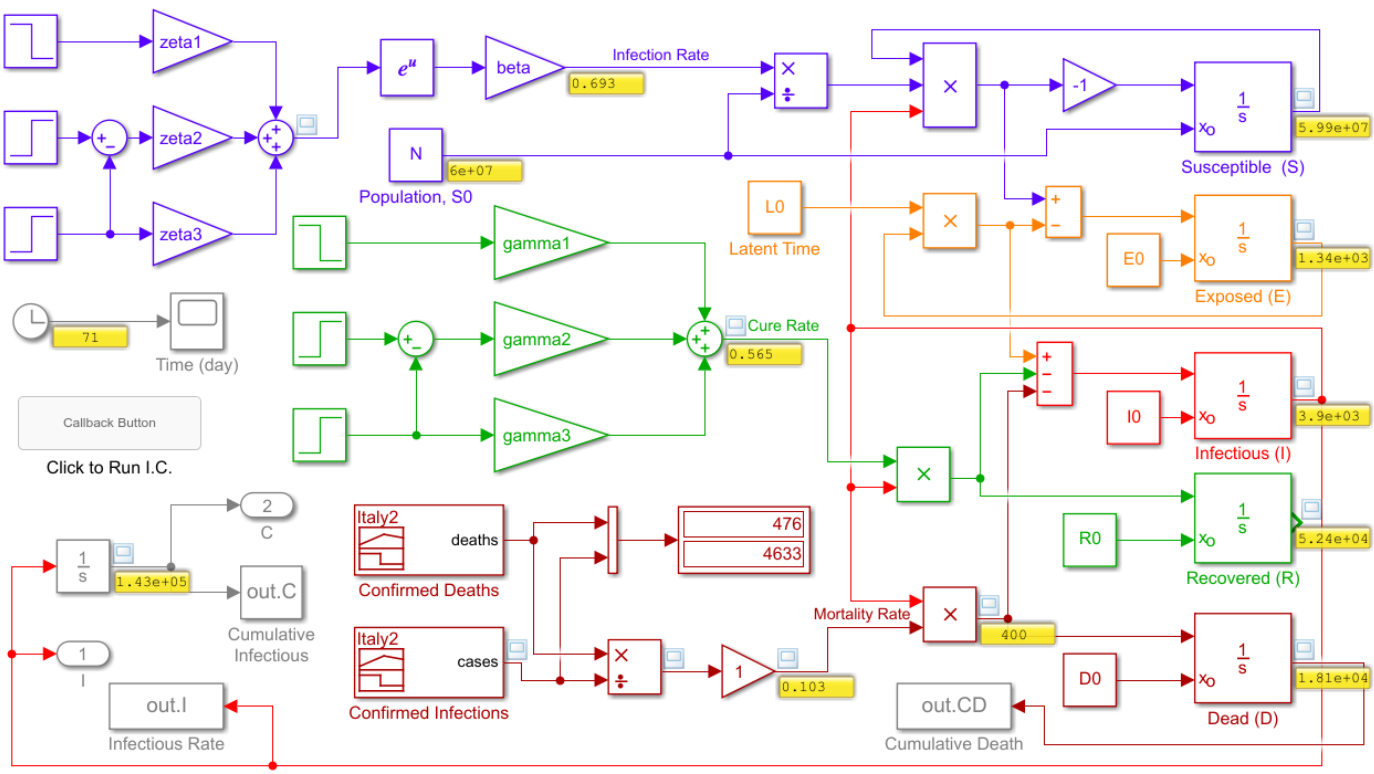

Fig. 6 SEIRD detailed model

-

5. Adaptive Neuro-Fuzzy SIR Model

The model used in the previous section was based on the mathematical model of the problem. We can also build a machine-learning program to simulate the same system using the input and output data of the model. Simulink provides the user with an adaptive neuro-fuzzy inference system (ANFIS) toolbox to generate if-then fuzzy rules automatically based on training the given data. References [18] present a detailed description of this technique in which the same methodology is employed in this paper. Using a basic SIR model built in Simulink with a variable infection rate and constant recovery rate, the model is trained using input-out data. The input could be the infection rate, recovery rate, or a combination of the two. The output could be the infectious output or its cumulative function. In this study, the beta function and its derivative are used as input variables to the ANFIS model whereas the infectious and cumulative infectious variables are chosen for the output in two different training processes. ANFIS allows us to use only one output for each block and for this reason, two separate processes are used to generate two ANFIS blocks for the two outputs. More outputs require more ANFIS blocks.

Figure 7 shows a simple Simulink program used in this training. The recovery function is treated as constant as proposed in [6] whereas the infection rate is treated as variable [6]. With the parameters optimized, the ANFIS toolbox is used to generate seven fuzzy rules for each output. The membership function used in this training is the gaussian function. Figure 8a-c shows the training iterations, fuzzy rules, and the ANFIS outputs for the cumulative and infectious variables. Figure 9 shows the Simulink model for the ANFIS blocks used in this simulation. These if-then rules are used to simulate the case of China outbreak and the results are shown in Fig. 10a. whereas Fig. 10b shows the infections and recovery parameter values. Notably, the results show some good matching but it needs improvement. A more detailed and complicated beta function formula can be used to improve the accuracy of the results.

6. Conclusion

This paper presents an open-source program to model, simulate, and predict the coronavirus COVID-19 outbreak using Simulink. Several models were employed including a simple SIR representation with constant recovery rate and a one-step beta function, SIR with three step-functions, and a more detailed SEIRD model. In addition, an adaptive neuro-fuzzy inference system was used to generate the output of the model based on some training tasks applied to the system. The tests used in this study were the cases of China and Italy outbreaks. The results obtained showed good matching between the simulated and reported cases. It also estimates the case of Italy to reach around 265000 cases at the end of July 2020. This paper promises some lasting values in the field of the coronavirus spread. The program can be used as an educational tool or for research studies.

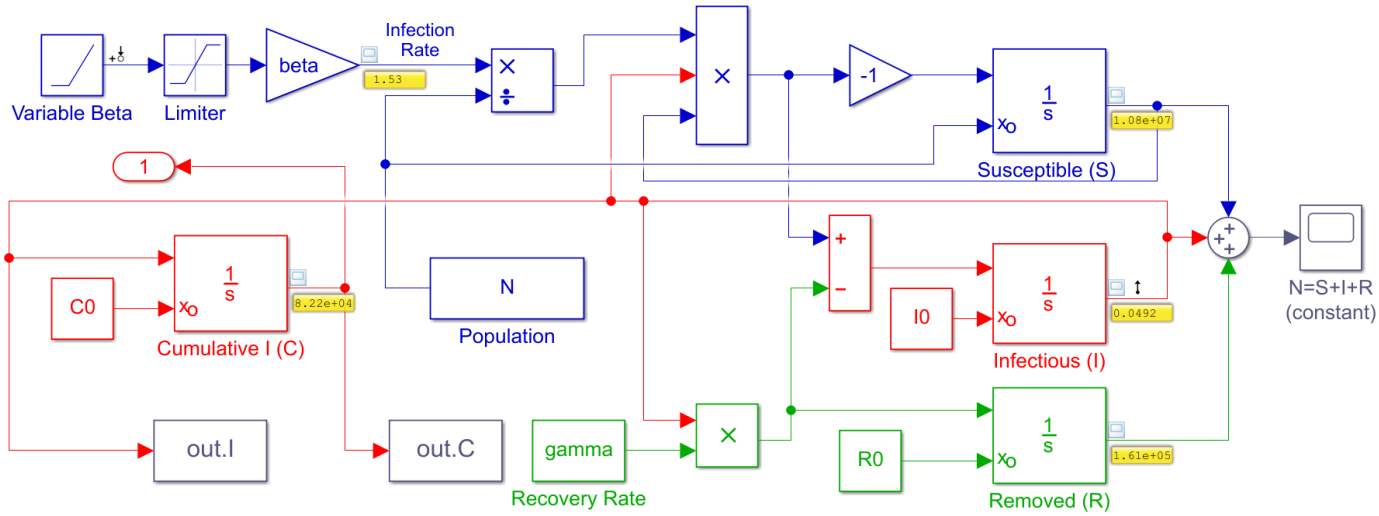

Fig. 7 A simple Simulink program used for building an ANFIS block

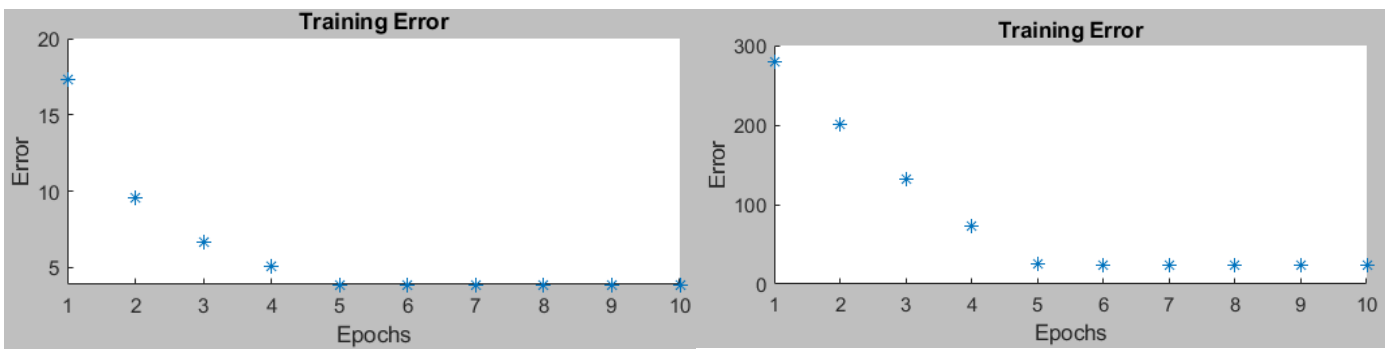

Fig. 8a Training errors for the infectious variable (left) and the cumulative infectious variable (right)

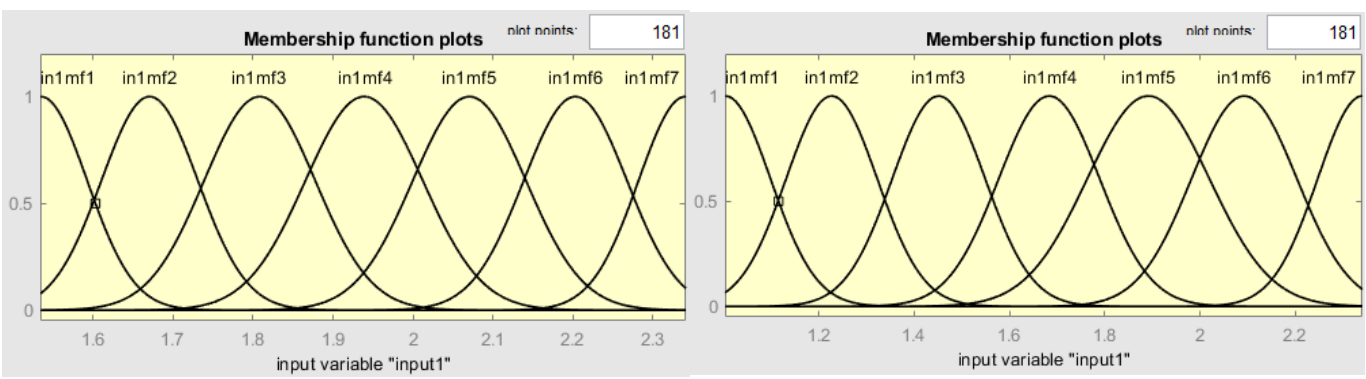

Fig. 8b Fuzzy rules generated by ANFIS for the infectious variable (left) and the cumulative infectious variable (right)

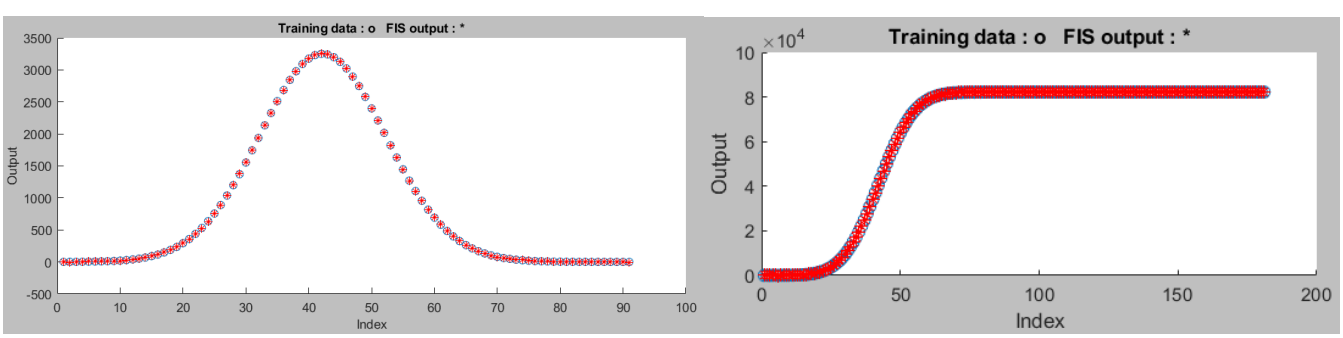

Fig. 8c Testing data against ANFIS output for the infectious variable (left) and the cumulative infectious variable (right)

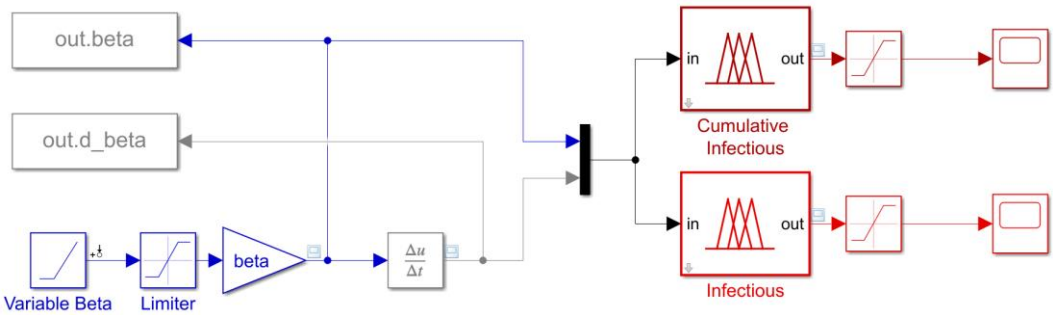

Fig. 9 The developed ANFIS model in Simulink

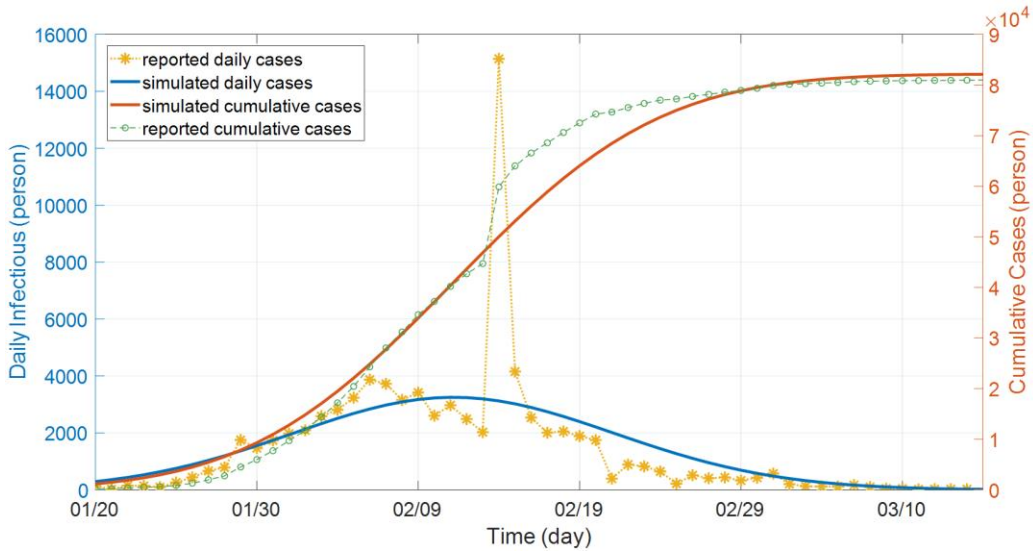

Fig. 10a ANFIS outputs and the reported cases (daily on the left y-axis and cumulative on the right y-axis) – case of China

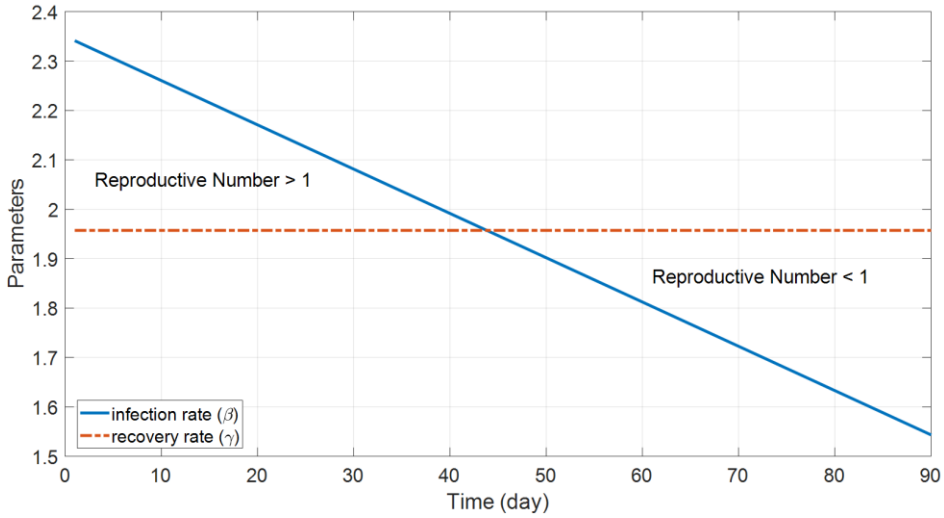

Fig. 10b Optimized parameters used in building the ANFIS model

Appendix I

The programs are available at:
https://www.mathworks.com/matlabcentral/fileexchange/75025-simcovid-an-open-source-simulation-program-for-the-covid-19

Appendix II

Demonstrating Videos

[1]. https://www.dropbox.com/s/sbnn1784a00hiw7/ANFIS.webm?dl=0

[2]. https://www.dropbox.com/s/mxbz8j9ogoom65g/China_Wuhan_SEIRD.webm?dl=0

[3]. https://www.dropbox.com/s/myi95d6dwnttoc8/ChinaOneStep.webm?dl=0

[4]. https://www.dropbox.com/s/mj6bi1jbiik9c03/Italy_SEIRD.webm?dl=0

[5]. https://www.dropbox.com/s/ncihp1dbqx1p35m/Italy_Optimization.webm?dl=0

[6]. https://www.dropbox.com/s/illjl95v4r603og/Italy.webm?dl=0

References

[1]. N. Richard, D. Robert , D. Valentin, H. Emma, and A. Jan, "Potential impact of seasonal forcing on a SARS-CoV-2 pandemic", Swiss Medical Weekly, March 2020,150:w20224. Available at: https://smw.ch/article/doi/smw.2020.20224

[2]. WHO Emergency Committee. Statement on the second meeting of the international health regulations (2005) emergency committee regarding the outbreak of novel coronavirus (2019-ncov). 2020. Available at: https://www.who.int/news-room/detail/30-01-2020-statement-on-the-second-meeting-of-the-international-health-regulations-(2005)-emergency-committee-regarding-the-outbreak-of-novel-coronavirus-(2019-ncov)

[3]. WHO Director-General's opening remarks at the media briefing on COVID-19. 2020. Available at: https://www.who.int/dg/speeches/detail/who-director-general-s-opening-remarks-at-the-media-briefing-on-covid-19--3-april-2020.

[4]. WHO Emergency Committee. Novel Coronavirus (2019-nCoV) Situation Report – 76. 2020. Available at: https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200405-sitrep-76-covid-19.pdf?sfvrsn=6ecf0977_2

[5]. Global Health NOW, Johns HopkinsBloomberg School of Publock Health, Available at: https://www.globalhealthnow.org/2020-04/1-million

[6]. L. Zhong, L. Mu1, J. Li, J. Wang, Z. Yin, and D. Liu, "Early Prediction of the 2019 Novel Coronavirus Outbreak in the Mainland China Based on Simple Mathematical Model", DOI: 10.1109/ACCESS.2020.2979599, pp. 51761 – 51769, Vol. 8, 2020. Available at: https://ieeexplore.ieee.org/document/9028194

[7]. I. Nesteruk, "Statistics Based Models for The Dynamics of Chernivtsi Children Disease", doi: DOI: 10.20535/1810-0546.2017.5.108577. Available at: http://bulletin.kpi.ua/article/view/108577

[8]. [9]D. Caccavo1, "Chinese and Italian COVID-19 outbreaks can be correctly described by a modified SIRD model" Pre-print manuscript, March 19, 2020. doi: 10.1101/2020.03.19.20039388. Available at: https://www.medrxiv.org/content/10.1101/2020.03.19.20039388v2

[9]. C. Anastassopoulou , L. Russo , A. Tsakris , C. Siettos, "Data-Based Analysis, Modelling and Forecasting of the Novel Coronavirus (2019-Ncov) Outbreak", medRxiv preprint doi: 10.1101/2020.02.11.20022186. Available at: https://www.medrxiv.org/content/10.1101/2020.02.11.20022186v5

[10]. M. Khan, A. Atangana, Modeling the dynamics of novel coronavirus (2019-nCov) with fractional derivative, Alexandria Engineering Journal, DOI: 10.1016/j.aej.2020.02.033. Available at: https://www.sciencedirect.com/science/article/pii/S1110016820300971

[11]. L. Peng et al, "Epidemic analysis of COVID-19 in China by dynamical modeling", Preprint arXiv:2002.06563v1. Available at: https://arxiv.org/abs/2002.06563v1?utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+CoronavirusArXiv+%28Coronavirus+Research+at+ArXiv%29

[12]. M. Batista, Estimation of the final size of the coronavirus epidemic by the SIR model, Updated on April 06, 2020. Available at: https://www.mathworks.com/matlabcentral/fileexchange/74658-fitviruscovid19

[13]. M. Batista, Estimation of the final size of the coronavirus epidemic by the logistic model, Updated on March 25, 2020. Available at: https://www.mathworks.com/matlabcentral/fileexchange/74411-fitvirus

[14]. COVID-19 Hospital Impact Model for Epidemics (CHIME), The Trustees of the University of Pennsylvania, 2020. Available at: https://penn-chime.phl.io/

[15]. J. CIAROCHI, How COVID-19 and Other Infectious Diseases Spread: Mathematical Modeling, March 12, 2020. Available at: https://triplebyte.com/blog/modeling-infectious-diseases

[16]. European Centre for Disease Prevention and Control, Situation update worldwide, as of 9 April 2020, Available at: https://www.ecdc.europa.eu/en/geographical-distribution-2019-ncov-cases
https://ieeexplore.ieee.org/document/8506642

[17]. Javier Montalt Tordera (2020). Filled area plot. Available at: https://www.mathworks.com/matlabcentral/fileexchange/69652-filled-area-plot), MATLAB Central File Exchange. Retrieved April 14, 2020.

[18]. I. Abdulrahman and G. Radman, "Wide-Area-Based Adaptive Neuro-Fuzzy SVC Controller for Damping Interarea Oscillations," in *Canadian Journal of Electrical and Computer Engineering*, vol. 41, no. 3, pp. 133-144, Summer 2018. Available at: https://ieeexplore.ieee.org/document/8506642