N-STEPS AHEAD OPTIMAL CONTROL OF A COMPARTMENTAL MODEL OF COVID-19

Douglas Madalena Martins

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Amit Bhaya

Rio de Janeiro
Janeiro de 2022

N-STEPS AHEAD OPTIMAL CONTROL OF A COMPARTMENTAL MODEL OF COVID-19

Douglas Madalena Martins

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Amit Bhaya, Ph. D.

Prof. Fernando Pazos, D. Sc.

Prof. Eugenius Kaszkurewicz, D. Sc.

Prof. Pierre-Alexandre Jacques Bliman, Dr.

RIO DE JANEIRO, RJ – BRASIL
JANEIRO DE 2022

Martins, Douglas Madalena

N-steps ahead optimal control of a compartmental model of COVID-19/Douglas Madalena Martins. – Rio de Janeiro: UFRJ/COPPE, 2022.

VII, 62 p.: il.; 29, 7cm.

Orientador: Amit Bhaya

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2022.

Referências Bibliográficas: p. 58 – 62.

1. System Identification. 2. Optimization. 3. Machine Learning. I. Bhaya, Amit. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Literature Review

The COVID-19 pandemic is an ongoing pandemic of coronavirus disease 2019 and has emerged as one of this century's major global health challenges. Insufficient scientific knowledge, the fast pace of its spread, and its capacity to cause deaths in vulnerable groups have generated worldwide discussion and research on the best strategies for confronting the epidemic in different parts of the world.

Governments are struggling to determine the correct course of action as the epidemic goes through its stages. If they decide to do nothing, a lot of deaths, mainly of the most susceptible people, will occur. On the other hand, full lockdowns affect the economy and society negatively. Modeling the pandemic also poses several difficulties, amongst these the rapid variation of several important parameters.

Although several vaccines have been developed, most countries have insufficient supplies to be able to vaccinate at a recommended level. Also, even though vaccines are effective against serious symptoms, they do not guarantee complete immunity so that strategies such as social distancing, washing hands and wearing face masks, known collectively as non-pharmaceutical interventions (NPIs), continue to play an important role in controlling this epidemic.

The most simple and commonly used model is the SIR model (first introduced in [2]) for human-to-human transmission, which describes the passage of individuals through three mutually exclusive stages of infection: Susceptible, Infected and Recovered ([3]). This chapter will limit its review to the recent literature on the modeling of COVID-19. For further references to the extensive literature on mathematical models for epidemics as well as its history the reader is directed to the books: Mathematical Epidemiology ([4]), Modeling Infectious Diseases in Humans and Animals ([5]), A Short History of Mathematical Population Dynamics ([6]).

Carcione et al. ([7]) implemented an SEIR model to compute the infected population and the number of casualties in the Italian region of Lombardy, one of the

regions most severely impacted by the epidemic in the world. The additional feature of this model, with respect to the SIR model, is the exposed state E, which represents people who have been exposed to the virus, but still not developed the infection, due to the incubation period of the virus. After this period, the exposed population transitions to the Infected state.

Giordano et al. ([3]) used a more complex model named SIDARTHE, that discriminates between detected and undetected cases of infection and between different severities of illness, non-life-threatening cases and potentially life-threatening cases that require ICU admission. The eight states are the following: S, the susceptible population; I, the asymptomatic undetected infected population; D, the diagnosed population, corresponding to asymptomatic detected cases; A, the ailing population, corresponding to the symptomatic undetected cases; R, the recognized population, corresponding to the symptomatic detected cases; T, the threatened population, corresponding to the detected cases with life-threatening symptoms; H, the hospitalized population; and E, the extinct or dead population. One of the difficulties with this model is the inaccurate estimation of some of the populations, such as populations A and I.

A model named SEIHRD was introduced in [8] and studied further in [1]. The main difference is the state Hospitalized corresponding to people who requires ICU installations. Inclusion of the hospitalized population in the model is important from a strategic point of view, since it allows public health officials to avoid shortages in hospital beds and supplies. In addition, the hospitalized population is a variable that is easy to monitor and is made available in real time. This model is the one that is used in this work and will be further explained in the next section.

After choosing an appropriate model, the next challenge is to use an algorithm that is able to control the epidemic to a certain level. Many authors have used different types of open-loop optimal control, and Model Predictive Control (MPC) has also been proposed as a closed loop control technique. It has been proven that an open-loop optimal control leads to simple policies under the assumption of exact model knowledge, but in a more realistic scenario with uncertain data and model mismatch, a feedback strategy that periodically updates the policy is much more effective, as stated in [9].

Bin et al. ([10]) proposed a fast switching policy, consisting of multi-shot interventions based on the outcomes of two SIR-based models (SIQR and SIDARTHE) to switch between quarantine (social isolation) and work days (normal behavior).

Pazos et al. ([1]) proposed a specific controller for determining the optimal intensity of the NPIs using a SEIHRD model. The values calculated are based on a proportional value to an adequate combination of states of this model. The control law applied was robust to relatively large parametric uncertainties and also to some

level of noncompliance of the NPIs.

A robust economic MPC for the containment of a generic stochastic SEIV (Susceptible-Exposed-Infected-Vigilant) epidemic process is presented in [11], with the final aim of deciding who to quarantine and for how long, in the presence of an epidemic contagion.

Some authors considered the influence of vaccination on the epidemic model. Kar et al. [12] studied a SIR epidemic model with a vaccination program. They used optimal control strategies in the form of vaccination to control the number of susceptible individuals and increase the number of recovered individuals.

An optimal daily vaccination strategy is proposed in [13]. They established an optimal control problem to design vaccination strategies where vaccination modulates dynamics susceptibility through an imperfect vaccine. The aim was to provide vaccination policies that minimize the lost life years due to disability or premature death by COVID-19. The simulations suggested a better response compared with a constant vaccination rate.

In contrast with most papers aiming at reproducing the dynamics of the pandemic observed through various data, a study related to the concepts of epidemic final size and herd immunity in an ample setting is done in [14]. They considered an epidemic in a heterogeneous population modelled by a SEIR system with a continuous structure variable and a general contact matrix. They derived and studied the final size equation fulfilled by the limit distribution of the population and showed that this limit exists and satisfies the final size equation. The main contribution was to prove the uniqueness of this solution among the distributions smaller than the initial condition.

Bliman et al. ([15]) investigated the effects of social distancing on a simple SIR model. They show that it is possible to exactly answer the following question: given maximal social distancing intensity and duration (but without prescribed starting date), how can one minimize the epidemic final size, that is the total number of individuals infected during the outbreak? They proved the existence of an unique optimal policy and demonstrated how to determine it numerically by an easily tractable algorithm.

An optimal control problem of obtaining, by enforcing social distancing, the largest value for the number of susceptible individuals at infinity is studied in [16]. They first established that stopping arbitrarily close to the herd immunity threshold through long enough intervention is possible only if the social distancing intensity is sufficiently intense. As a last result, they show that this problem may be interpreted as equivalent to reaching a given distance to the herd immunity level by minimal intervention time.

The main contributions of this dissertation are:

- A simplified MPC type control approach that reduces computational effort considerably, thus allowing the simulation of various scenarios.
- A normalized aggregate control effort that models the effect of all non-pharmaceutical interventions and therefore takes values between zero and one, rather than being on-off.
- Inclusion of the effect of different vaccination policies on the progress of the pandemic.
- A simulation platform that permits understanding and design of public health policies for the short and medium term control of the pandemic.

In this work, we investigate strategies based on N-Steps ahead optimal control for mitigation of the COVID-19 pandemic. The main goal is to minimize the number of deaths over time without inducing excessive economic costs, while respecting an upper bound on the hospitalization rate. In order to do that, an optimization problem is modeled and solved based on the techniques presented in [17] and [18].

In chapter 2, the epidemic model SEIHRD is explained in more details. In chapter 3, we detail the optimal algorithm that is used to calculate the best strategy for each moment of the epidemic. In chapter 4, we develop all simulations and compare our strategy with other strategies already used for the same problem. Additionally, the impact of the vaccination rate is shown in Chapter 5. Finally, conclusions and future work are presented in chapter 6.

Chapter 2

The SEIHRD Model

In this chapter, we will detail each state of the SEIHRD model and its relevance in the COVID-19 epidemic model. The choice of the model is an important step. In order to be useful for the design of control policies, it should contain the main variables of interest, keeping in mind the difficulty of obtaining reliable data that will permit estimation of the main model parameters. In this study, we opted for a model called the SEIHRD model, explained in the next paragraph, since it allows for a more detailed model of features specific to the COVID-19 epidemic, such as exposed and asymptomatic populations, in addition to modeling occupation of hospitals, which is important from a decision making perspective.

The SEIHRD model contains the following states or populations that each individual can belong to:

- Susceptible (S): Individuals who did not get exposed to the virus and are not infected.
- Exposed (E): Individuals who got exposed to the virus and are in the incubation period. Even though there are no visible clinical signs, the individual could infect other people with a lower probability (compared to one in the Infected state). Part of this group will present symptoms after an incubation period, moving to group I and another part will remain asymptomatic.
- Infected (I): Individuals who can infect other people and may start developing clinical signs. Asymptomatic people who have been diagnosed as positive are also considered in this group. After a period, the individual recover or is hospitalized, if the symptoms are very serious.
- Hospitalized (H): Individuals who need medical assistance and occupy beds in the hospital. After treatment, the individual might recover or die.
- Recovered (R): Individuals who recover from the infection or acquired immunity.

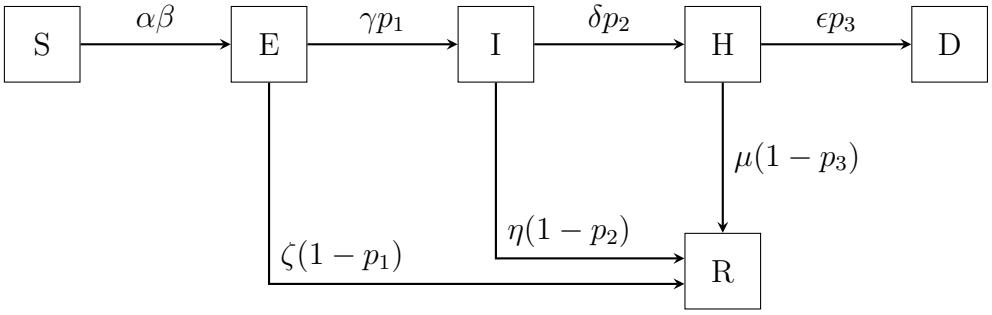

Figure 2.1: SEIHRD model diagram.

- Dead (D): Individuals who were infected, hospitalized and then died.

This is a typical compartmental model and Fig. 2.1 shows the manner in which individuals transit between these states or populations. This model also has a mathematical representation given by the following difference equations.

$$S_{k+1} = S_k - (\alpha S_k E_k + \beta S_k I_k) \tag{2.1}$$

$$E_{k+1} = E_k + (\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1-p_1))E_k \tag{2.2}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1-p_2))I_k \tag{2.3}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1-p_3))H_k \tag{2.4}$$

$$R_{k+1} = R_k + \zeta(1-p_1)E_k + \eta(1-p_2)I_k + \mu(1-p_3)H_k + \upsilon_k \tag{2.5}$$

$$D_{k+1} = D_k + \epsilon p_3 H_k \tag{2.6}$$

where:

- $k \in \{1, 2, ..., K\}$ where $K \in \mathbb{N}$ is the maximum time horizon considered in the study.
- $\alpha S_k E_k$ is the transmission rate of the virus between Susceptible and Exposed, while $\beta S_k I_k$ is the transmission between Susceptible and Infected. Parameters α and β are the probability of disease transmission in a single contact per person. Typically, α is greater than β, since each individual tend to avoid contact with people showing symptoms. Also, the viral load is higher in the second case.
- p_1 is the probability that exposed people develop symptoms.
- γ^{-1} is the average period to develop symptoms.
- ζ^{-1} is the average time to overcome the disease while remaining asymptomatic.

- p_2 is the probability that infected people with symptoms require hospitalization.
- δ^{-1} is the average time between infection and the need for hospitalization.
- η^{-1} is the average time for infected people to recover without hospitalization.
- p_2 is the probability that infected people with symptoms required hospitalization.
- ϵ^{-1} is the average time between the hospitalization and death.
- μ^{-1} is the average time to recover after hospitalization.
- p_3 is the probability that hospitalized people die.

According to the equations, the populations in the compartments R and D are always increasing, while S is always decreasing. This is expected, since the number of Recovered and Dead people may stop increasing but will never decrease (with the assumption that reinfections are not possible). The same idea can be applied to the group S, that will decrease until the epidemic is finished.

This model does not discriminate detected and undetected cases of infection as this would add an extra complexity. Although it ignores the more complex biology, it does allow the inclusion of some important real world issues such as scarcity of hospital beds.

The transference between the model compartments is based on mean rates, indicating that the individuals stay for a certain period in each compartment. This could also be represented by adding delays instead of using mean rates. There are other modeling techniques, like in [19] which uses a conveyor to represent the delays. However, in this work we chose to use the mean rates as a simplification.

A basic quantity in the analysis of epidemic models is the basic reproduction number R_0, which, informally, is the expected number of people who will be infected by one person with the disease. If R_0 is less than 1, each infected person can transmit the virus to less than one susceptible person. This means the number of infected will decline and the disease will die out. If R_0 is greater than 1, the disease will spread into the population and the number of infected people will increase, causing an epidemic. A detailed explanation of the basic reproduction number can be found in [2].

The parameters used in the model might assume different values for different regions in the world. Specially, the parameters α and β are related with R_0 and they are influenced by different factors, like population density of a community, the general health and average age of its population, medical infrastructure.

Country	Median R_0	Confidence interval - R_0
Belgium	3.6	(2.9, 4.6)
France	4.4	(3.6, 5.4)
Germany	4.7	(3.8, 5.8)
Italy	4.6	(3.7, 5.8)
Netherlands	3.5	(3.0, 4.2)
Spain	6.4	(5.2, 8.0)
Switzerland	3.5	(2.8, 4.3)
United Kingdom	3.9	(3.3, 4.6)
United States	5.9	(4.7, 7.5)

Table 2.1: R_0 values for European countries and United States.

Parameter	Value
α	0.179
β	0.0895
γ^{-1}	5.1
ζ^{-1}	14.7
δ^{-1}	5.5
η^{-1}	14
ϵ^{-1}	11.2
μ^{-1}	16
d_1	21
p_1	50%
p_2	19%
p_3	15%

Table 2.2: Parameter values used in the SEIHRD model (based on [1]).

Ke et al. [20] collected data from the United States and eight countries from Europe before control measures were implemented during March 2020. They show that COVID-19 has high R_0 values and spread very rapidly in the absence of strong control measures across different countries. This implies very high herd immunity thresholds and highly effective vaccines with high levels of population coverage will be needed to prevent sustained transmission. The results are illustrated in table 2.1.

In this work, the values of the parameters used in (2.1)-(2.6) are based on the studies made in [1] and can be found in table 2.2.

Chapter 3

Optimal Control Problem

In this chapter, we first present the theory behind an optimal control problem. We base our studies mainly on [18] and [17]. In the subsequent sections, we introduce and explain different types of controller and strategies used to control the COVID-19 epidemic.

Consider a dynamical system described by the following difference equation:

$$x_{i+1} - x_i = f_i(x_i, u_i), \quad i = 0, 1, ..., k-1, \tag{3.1}$$

where $x_i \in E^n$ is the state of the system at time i, $u_i \in E^m$ is the input to the system at time i, and $f_i(\cdot, \cdot)$ is a function mapping $E^n \times E^m$ into E^n. The optimal control problem consists in finding the a control sequence $\hat{\mathfrak{u}} = (\hat{u_0}, \hat{u_1}, ..., \hat{u_{k-1}})$ and a corresponding trajectory $\hat{\mathfrak{X}} = (\hat{x_0}, \hat{x_1}, ..., \hat{x_k})$ determined by (3.1), which minimize the sum:

$$\sum_{i=0}^{k-1} f_i^0(x_i, u_i), \tag{3.2}$$

where the f_i^0 map $E^n \times E^m$ into $\mathbb{R}$.

This minimization is subject to the following constraints, which we write as the intersection of inequality and equality constraints whenever appropriate. The control constraints are:

$$u_i \in U_i \subset E^m \quad i = 0, 1, \cdots, k-1. \tag{3.3}$$

The initial boundary constraints are:

$$x_0 \in X_0 = X_0' \cap X_0'' \quad X_0' = \{x : q_0(x) \leq 0\}, X_0'' = \{x : g_0(x) = 0\}, \tag{3.4}$$

where $q_0(\cdot)$ maps E^n into E^{m_0} and $g_0\cdot$ maps E^n into E^{l_0}. The terminal boundary

constraints are:

$$x_k \in X_k = X'_k \cap X''_k \quad X'_k = \{x : q_k(x) \leq 0\}, X''_k = \{x : g_k(x) = 0\}, \tag{3.5}$$

where $q_k(\cdot)$ maps E^n into E^{m_k} and $g_k\cdot$ maps E^n into E^{l_k}. The state-space constraints are:

$$x_i \in X_i = X'_i \cap E^n = \{x : q_i(x) \leq 0\} \quad i = 1, 2, \cdots, k-1, \tag{3.6}$$

where $q_i(\cdot)$ maps E^n into E^{m_l}. This problem may be recast in the form:

$$\begin{aligned} &\text{minimize} && f(z) \\ &\text{subject to:} && r(z) = 0 \end{aligned} \tag{3.7}$$

where $z \in \Omega$ and the following identifications are made:

$$f(z) = \sum_{i=0}^{k-1} f_i^0(x_i, u_i) \tag{3.8}$$

$$r(z) = \begin{bmatrix} x_1 - x_0 - f_0(x_0, u_0) \\ \cdots \\ x_k - x_{k-1} - f_{k-1}(x_{k-1}, u_{k-1}) \\ g_0(x_0) \\ g_k(x_k) \end{bmatrix} \tag{3.9}$$

$$\Omega = X'_1 \times X'_1 \times X'_2 \times \cdots \times X'_{k-1} \times X'_k \times U_0 \times U_1 \times \cdots \times U_{k-1} \tag{3.10}$$

3.1 SEIHRD model with control variable

The characteristics of COVID-19 disease make the virus spread incredibly fast into the population. One of the strategies the government can apply is to attack the source, when the disease is not yet disseminated in the population. Isolation of cases and tracking of new cases and people that were in contact with someone infected would interrupt the transmission in the source. Of course, this is very difficult in a globalized world, where people can easily travel to all places.

Another strategy is to interrupt (or reduce) the transmission. This is mainly achieved by increasing personal and environmental hygiene (washing hands, etc.), using appropriate masks when in public and restricting population movements. A

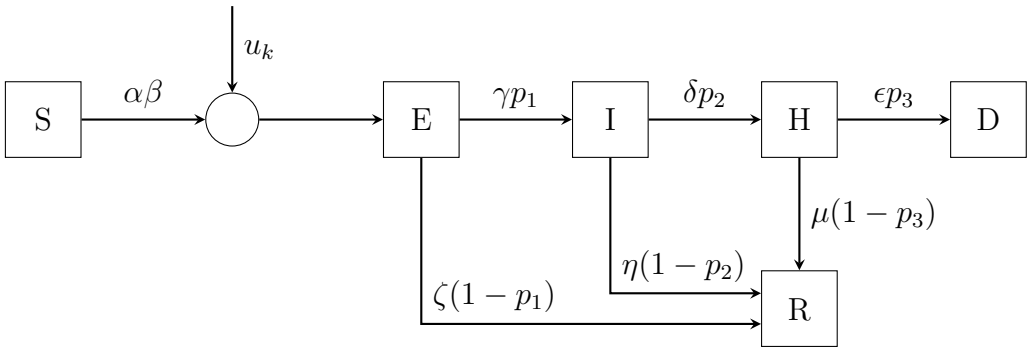

Figure 3.1: SEIHRD model diagram with control variable.

lockdown strategy and social distancing have been observed in several parts of the world. Middle and low income countries do not have sufficient resources and face financial and economic challenges, which may hinder their ability to effectively implement the above mentioned policies.

The strategies described above are the so called Non Pharmaceutical Interventions (NPIs). Pharmaceutical intervention is mainly achieved by vaccinating the population since, to date, there is no proven reliable treatment for infected people.

In this work, an aggregate normalized control effort varying between zero and one will be taken to represent all the NPIs being applied (i.e., lumping together social distancing, use of masks, adoption of hygienic measures, etc.).

Since we can only try to control the transmission between individuals, in the mathematical model introduced in chapter 2 we may only affect the relations between compartmental groups S, E and I. In other words, we would like to prevent susceptible people from getting into contact with people that have the virus (Exposed and Infected). Therefore, the proposed model is the following:

$$S_{k+1} = S_k - (1-u_k)(\alpha S_k E_k + \beta S_k I_k) \tag{3.11}$$

$$E_{k+1} = E_k + (1-u_k)(\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1-p_1))E_k \tag{3.12}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1-p_2))I_k \tag{3.13}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1-p_3))H_k \tag{3.14}$$

$$R_{k+1} = R_k + \zeta(1-p_1)E_k + \eta(1-p_2)I_k + \mu(1-p_3)H_k + v_k \tag{3.15}$$

$$D_{k+1} = D_k + \epsilon p_3 H_k \tag{3.16}$$

The control variable affects directly the parameters α and β of the SEIHRD model, as shown in Figure 3.1. One important fact is that by increasing the control input, we are decreasing the observed α and β. In other words, the control prevents susceptible individuals to get in contact with exposed and infected individuals, respectively.

The control variable u_k can assume any value between 0 and 1 ($u_k \in [0, 1]$). The lower bound value 0 means that no social distancing strategy is applied and people are free to go wherever they want. The upper bound value 1 means that a lockdown is in place and people have no contact with each other, meaning that the transmission is interrupted. This is impossible in practice, since basic services for the population require some movement of populations.

It should be observed that states D and R do not affect the dynamics of the rest of the model (i.e., do not occur in the equations (3.11)-(3.14)). In the next sections, we will not include them in the optimization problem, since they would only add unnecessary complexity. However, they can be calculated using the other state variable values.

Social distancing is the main NPI strategy to interrupt the spread of the virus in the population. When the number of infected and hospitalized people is too high, social distancing needs to be implemented. Since complete lockdown has well known adverse effects in the economy, this work postulates a certain level of normalized control effort (between zero and one) that translates into partial lockdown and relaxation of other measures (such as the use of masks). The focus of this work is on strategies to calculate the value of this aggregate control. It is then the task of decision makers to translate this level of control effort into concrete NPI policies, which is, of course, a nontrivial task. In the next sections we will show different algorithms to calculate the best values of control variable u_k during the time horizon.

3.2 N-Steps Ahead Optimal Control (NSAOC)

In this work, we introduce a controller named N-Steps Ahead Optimal Control (NSAOC). The main idea is to calculate, at each time instant, a new control value based on the estimation of the evolution of the state vector during the next N time instants. So, at each time step $k = 1, 2, ..., K$, where k corresponds to days and K is the time horizon applied to the epidemic model, we solve an optimization problem over the horizon $k, ..., k + N$, where N is the number of days that we will use to calculate the best values for the control variable u. The optimization result will give us the best values for the next N control inputs. However, we will only use the entry k, since at instant $k + 1$ we will have more information from the real environment (the states of day $k + 1$ resulting from the applicability of u_k).

It is worth mentioning that the optimization problem does the calculations based on a certain epidemic model (SEIHRD in this work). However, the control variable is applied to a real environment, where some parameters (if not all) can differ from those of the model. This controller is a model predictive controller (MPC) with the difference that there are no uncertainty in the dynamics, only in the model

parameters.

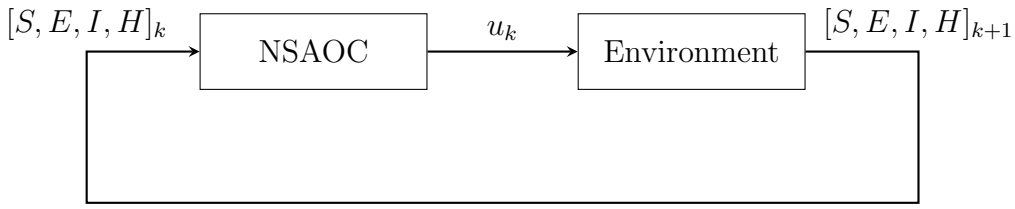

Figure 3.2: N-Steps Ahead Control Diagram.

The diagram in Fig. 3.2 illustrates the flow used in the problem. The optimization block (NSAOC) is responsible for the calculation of the next control inputs taking into consideration an adequate objective function. The mathematical model is described below:

$$\text{min} \qquad J = \sum_{k}^{k+N} u_k \tag{3.17}$$

$$\text{subject to:} \quad S_{k+1} = S_k - (1 - u_k)(\alpha S_k E_k + \beta S_k I_k) \tag{3.18}$$

$$E_{k+1} = E_k + (1 - u_k)(\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1 - p_1))E_k \tag{3.19}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1 - p_2))I_k \tag{3.20}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1 - p_3))H_k \tag{3.21}$$

$$H_k \leq H_{max} \tag{3.22}$$

$$0 \leq u_k \leq 1 \tag{3.23}$$

S_k, E_k, I_k and H_k are the initial conditions and input of the algorithm. H_{max} is the maximum capacity of the medical resources. If H_k assume values higher than it, the number of deaths would have a considerable increase, since the medical capacity will be exceeded and, consequently, part of the population will not be covered by medical care. No government wants this to happen, so the constraint (3.22) is added to make sure the optimization takes this into consideration.

The objective function (3.17) takes into account only resources available for the implementation of control efforts, while the constraints model available hospital infrastructure. In other words, (3.17) is an administrator's ideal objective function. In practice, of course, other humanitarian concerns, such as limiting the number of deaths, are more important and should also be added to the objective function. The formulation presented in this dissertation is applicable to all such objective functions and can be regarded as a tool to aid decision making by simulating scenarios, with different objective functions, parameters and so on. Thus there are other objective

functions that might be tested as well. One can try to maximize the number of susceptible people and minimize the sum of all control effort, for example. Also, some weights can be added to each factor to give more importance to one or another.

Another important issue is the choice of suitable values for N. We would like to choose values as small as possible so we do not do extra calculations. On the other side, N cannot be too small, because, over a prediction horizon that is too short, it would fail to predict an exponential rise in hospitalizations and the situation would get out of control. In the next section we study what are the most suitable values for N, studying their impact in the environment.

The full algorithm we will use in this work is given below:

Algorithm 1: N-Steps Ahead Optimal Control Algorithm

for $k = 1 : K$ **do**

1) Solve (3.17) - (3.23) for the interval $(k : k + N)$, where N is the number of steps ahead used on the algorithm. In this stage, all states are estimated based on a COVID-19 Model with parameters defined in table 2.2;

2) The resultant control variable array has size N. Pick the first value of the array and use as input on the real environment, in which the parameters will most probably differ from the ones used in the model in Step 1;

3) Acquire (measure) the resultant state variables at instant $k + 1$ from the Environment and store them for use in next iteration.

end

Menezes Morato et al. ([21]) designed an optimal On-Off social isolation strategy based on a Model Predictive Control (MPC) policy. In contrast, as argued above, we allow variation in the isolation level, as the values can be chosen in the interval $[0, 1]$.

3.3 PID-Like Control

This section will briefly present a controller proposed in [1] for the purposes of comparison. They used control theory to determine public NPIs in order to control the evolution of the pandemic, avoiding the collapse of the health care systems while minimizing harmful effects on the population and economy.

Again, the control law is given by the control variable u_k in equations (3.11) - (3.16). No interventions is represented by $u_k = 0$ and a full lockdown with no movement allowed translates to $u_k = 1$.

There are several possible choices of the reference signal or set point of the control system. We also must keep in mind that some groups of the SEIHRD model are

subjected to large inaccuracies due to unreported or undiagnosed cases. So, ideally the controller should use states for which reliable data are available. The number of hospitalized (H) people is very reliable. The number of people diagnosed as positive (I) and deaths (D) are also reasonably reliable.

The controller proposed in [1] is shown schematically in Fig. 3.3. The set point (SP) represents parameter H_{max} explained in the previous section and specified in (3.22).

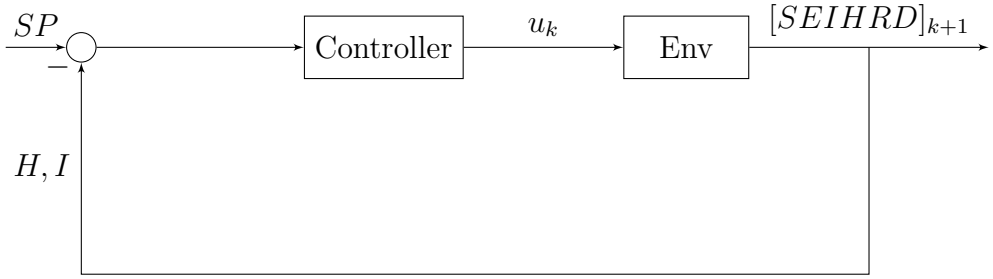

Figure 3.3: PID-Like Controller diagram ([1])

The obvious choice of feedback variable would be the number of hospitalized people. However, since NPIs reduce contagion between susceptible and infected or exposed people, when an individual is infected, hospitalization may be required after $\delta^{-1} = 5.5$ days or after $\delta^{-1} + \gamma^{-1} = 10.6$ days on average if the infection was recent. Hence, there exists a delay between the adoption of NPIs and their consequences on hospitalization. If the control action relies only on the number of hospitalized people, too many people may require hospitalization in the next 10.6 days, exceeding the capacity for medical care.

Therefore, the control action should also take into consideration the number of infected people. The addition of this state emulates a type of predictive control, since it is proportional to the number of people who would require hospitalization in the next five or six days.

However, not all infected people need hospitalization. It is reported that most symptomatic cases are mild and remain mild. According to table 2.2, $p_2 = 19\%$ of infected people will need hospitalization in the following 5.5 days (δ^{-1}). This number plus the number of people already hospitalized must remain below the set point. So, the normalized PID-Like control law proposed is:

$$u_k = k_p \left(1 - \frac{SP - H_k - p_2 I}{SP - H_k}\right) \in [0, 1] \tag{3.24}$$

where k_p is a scalar gain with values between $[0, 1]$.

3.4 Reinforcement Learning

The term "optimal control" came into use in the late 1950s to describe the problem of designing a controller to minimize or maximize a measure of a dynamical system's behavior over time. One of the approaches to this problem was developed in the mid-1950s by Richard Bellman and others by extending a nineteenth century theory of Hamilton and Jacobi. This approach uses the concepts of a dynamical system's state and of a value function, or "optimal return function", to define a functional equation, now often called the Bellman equation. The class of methods for solving optimal control problems by solving this equation came to be known as dynamic programming ([22]). Bellman ([23]) also introduced the discrete stochastic version of the optimal control problem known as Markov decision processes (MDPs). All of these are essential elements underlying the theory and algorithms of modern reinforcement learning.

Reinforcement learning is the problem faced by an agent (the learner) that needs to learn the right behavior through trial-and-error interactions with a dynamic environment. The agent is not told which actions to take, but instead must discover which actions yield the most reward by trying them. In some cases, the rewards are not immediate and the actions can affect the next situations and, consequently, all subsequent rewards.

Reinforcement learning is different from supervised learning. In general, the latter refers to learning from a training set of labeled examples provided by a knowledgeable external supervisor. This is not the case with reinforcement learning. Instead, it is necessary for the agent to gather useful experience about the possible system states, actions, transitions and rewards actively to act optimally.

The most important feature is that it uses training information that evaluates the actions taken rather than instructs by giving correct actions. This is what creates the need for active exploration, for an explicit search for good behavior.

The main components of reinforcement learning are the following:

- Agent: The entity that will interact with an environment via a policy.
- Environment: The dynamical system the agent interacts with.
- State: Defines the actual stage of the environment. When the agent selects an action, the state of the environment will change and the agent will gain a reward based on that.
- Policy: Defines the learning agent's way of behaving at a given time. A policy is a mapping from perceived states of the environment to actions to be taken when in those states.

- Reward Signal: Defines the goal of a reinforcement learning problem. At each time step, the environment sends to agent a single number called reward. The agent's sole objective is to maximize the total reward it receives over the long run.

- Value function: Specifies what is good in the long run, while reward signal indicates what is good in an immediate sense. Rewards are given directly by the environment, but values must be estimated from the sequences of observations an agent makes over its entire lifetime.

- Environment model: This is an optional component. An environment model mimics the behavior of the real environment. Given a state and action, the model might predict the resultant next state and next reward. Some methods (model-free methods) do not use it and are explicitly trial-and-error methods.

As in all of artificial intelligence, there is a tension between breadth of applicability and mathematical tractability. In order to formalize the problem of reinforcement learning, we need to visit the theory of Markov Decision Processes (MDPs).

3.4.1 Markov Decision Process

Markov Decision Processes (MDPs) are a classical formalization of sequential decision making, where actions influence not just immediate rewards, but also subsequent situations, or states, and through those future rewards. MDPs are a mathematically idealized form of the reinforcement learning problem for which precise theoretical statements can be made. The theory presented in this section is based on [24].

MDPs are meant to be a straightforward framing of the problem of learning from interaction to achieve a goal. The learner and decision maker is called the agent. The thing it interacts with, comprising everything outisde the agent, is called the environment.

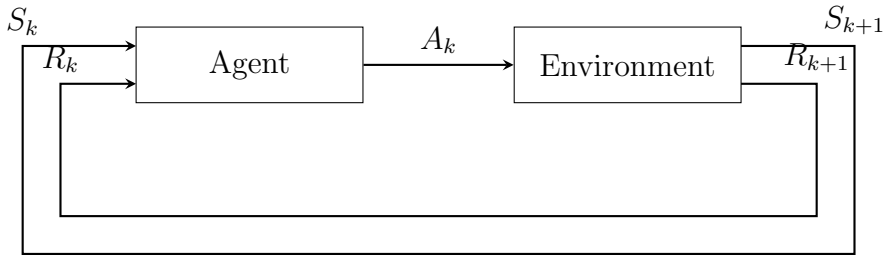

Figure 3.4: Reinforcement Learning framework.

At each time step k, the agent receives some representation of the environment's state ($S_k \in \mathcal{S}$). Based on that, an action $A_k \in \mathcal{A}(s)$ is selected to be applied on the environment. One time step later, the agent receives a numerical reward ($R_{k+1} \in \mathcal{R}$) as a consequence of its action and finds itself in a new state S_{k+1} (Figure 3.4). The MDP and agent give rise to a sequence or trajectory that begins like:

$$S_0, A_0, R_1, S_1, A_1, R_2, S_2, A_2, R_3, ... \tag{3.25}$$

In a finite MDP, the sets of states, actions and rewards($\mathcal{S}$, $\mathcal{A}$ and $\mathcal{R}$) have a finite number of elements. In this case, the random variables R_k and S_k have well defined discrete probability dependent only on the preceding state and action. That is, for particular values of these random variables, $s' \in \mathcal{S}$ and $r \in \mathcal{R}$, there is a probability of those values occurring at instant k, given particular values of the preceding state and action:

$$p(s', r \mid s, a) = Pr\{S_k = s', R_k = r \mid S_{k-1} = s, A_{k-1} = a\}, \tag{3.26}$$

for all s', $s \in \mathcal{S}$, $r \in \mathcal{R}$, $a \in \mathcal{A}(s)$. The function p defines the dynamics of the MDP. The probability of each possible value for S_k and R_k depends only on the immediately preceding state and action, S_{k-1} and A_{k-1}. The state must include information about all aspects of the past agent-environment interaction that make a difference for the future. If it does, then the state is said to have the Markov property.

In reinforcement learning, the purpose or goal of the agent is formalized in terms of a special signal, called the reward, passing from the environment to the agent. At each instant, the reward is a simple number, $R_k \in \mathbb{R}$. Informally, the agent's goal is to maximize the total amount of reward it receives. This means maximizing not immediate reward but cumulative in the long run. In general, we seek to maximize the expected return, where the return, denoted G_k, is defined as some specific function of the reward sequence. In the simplest case the return is the sum of the rewards:

$$G_k = R_{k+1} + R_{k+2} + R_{k+3} + ... \tag{3.27}$$

Equation (3.27) might be changed to add the concept of discounting. According to this approach, the agent tries to select actions so that the sum of the discounted rewards it receives over the future is maximized. In particular, it chooses A_k to maximize the expected discounted return:

$$G_k = R_{k+1} + \gamma R_{k+2} + \gamma^2 R_{k+3} + ... = \sum_{m=0}^{\infty} \gamma^m R_{k+m+1}, \tag{3.28}$$

where γ is a parameter called the discount rate ($0 \leq \gamma \leq 1$).

This approach makes sense in applications in which there is a natural notion of final time step, that is, when the agent-environment interaction breaks naturally into sub-sequences, which we call episodes. Each episode ends in a special state called the terminal state, followed by a reset to a standard starting state.

Almost all reinforcement learning algorithms involve estimating value functions, that estimate how good it is for the agent to be in a given state. The rewards the agent can expect to receive in the future depend on what actions it will take. Accordingly, value functions are defined with respect to particular ways of acting, called policies. A policy is a mapping from states to probabilities of selecting each possible action. If the agent is following policy π at instant k, then $\pi(a \mid s)$ is the probability that $A_k = a$ if $S_k = s$

The value function of a state

$$v_\pi(s) = \mathbb{E}_\pi[G_k \mid S_k = s] = \mathbb{E}_\pi \left[\sum_{m=0}^{\infty} R_{k+m+1} \mid S_k = s \right], \tag{3.29}$$

for all $s \in \mathcal{S}$, where $\mathbb{E}_\pi[\cdot]$ denotes the expected value of a random variable given that the agent follows policy π. We call the function v_π the state-value function for policy π.

Similarly, we define the value of taking action a in state s under a policy π, denoted $q_\pi(s, a)$, as the expected return starting from s, taking the action a and thereafter following policy π. We call q_π the action-value function for policy π.

$$q_\pi(s, a) = \mathbb{E}_\pi[G_k \mid S_k = s, A_k = a] = \mathbb{E}_\pi \left[\sum_{m=0}^{\infty} \gamma^m R_{k+m+1} \mid S_k = s, A_k = a \right]. \tag{3.30}$$

A fundamental property of value functions used throughout reinforcement learning and dynamic programming is that they satisfy recursive relationships. For any policy π and any state s, the following consistency condition holds between the value of s and the value of its possible successor states:

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_\pi[G_k \mid S_k = s] \\ &= \mathbb{E}_\pi[G_{k+1} + \gamma G_{k+1} \mid S_k = s] \\ &= \sum_a \pi(a \mid s) \sum_{s'} \sum_r p(s', r \mid s, a) \left[r + \gamma \mathbb{E}_\pi \left[G_{k+1} \mid S_{k+1} = s' \right] \right] \\ &= \sum_a \pi(a \mid s) \sum_{s', r} p(s', r \mid s, a) \left[r + \gamma v_\pi(s') \right], \end{aligned} \tag{3.31}$$

for all $s \in \mathcal{S}$, where it is implicit that the actions, a, are taken from the set $\mathcal{A}(s)$,

that the next states, s', are taken from the set $\mathcal{S}$, and the rewards, r, are taken from the set $\mathcal{R}$. Equation (3.31) is the Bellman equation for v_π. It expresses a relationship between the value of a state and the values of its successor states.

Solving a reinforcement learning task means, roughly, finding a policy that achieves a lot of reward over the long run. For finite MDPs, a policy π is defined to be better than or equal to a policy π' if its expected return is greater than or equal to that of π' for all states. In other words, $\pi \geq \pi'$ if and only if $v_\pi(s) \geq v_{\pi'}(s)$ for all $s \in \mathcal{S}$. There is always at least one policy that is better than or equal to all other policies. This is an optimal policy. It might exist more than one optimal policy and they are denoted by π_*. They share the same state-value function, defined as:

$$v_*(s) = \max_\pi v_\pi(s), \tag{3.32}$$

for all $s \in \mathcal{S}$.

Optimal policies also share the same optimal action-value function, denoted q_*, and defined as:

$$q_*(s,a) = \max_\pi q_\pi(s,a), \tag{3.33}$$

for all $s \in \mathcal{S}$ and $a \in \mathcal{A}(s)$. We can also write q_* in terms of v_*:

$$q_*(s,a) = \mathbb{E}\left[R_{k+1} + \gamma v_*(S_{k+1}) \mid S_k = s, A_k = a\right]. \tag{3.34}$$

Function v_* is the value function for a policy, so, consequently, it needs to satisfy the self-consistency condition given by the Bellman equation for state values (3.31). The Bellman equation for v_* is expressed in the following way:

$$\begin{aligned} v_*(s) &= \max_{a \in \mathcal{A}(s)} q_{\pi_*}(s,a) \\ &= \max_a \mathbb{E}_{\pi_*}\left[G_k \mid S_k = s, A_k = a\right] \\ &= \max_a \mathbb{E}_{\pi_*}\left[R_{k+1} + \gamma G_{k+1} \mid S_k = s, A_k = a\right] \\ &= \max_a \mathbb{E}\left[R_{k+1} + \gamma v_*(S_{k+1}) \mid S_k = s, A_k = a\right] \\ &= \max_a \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma v_*(s')\right] \end{aligned} \tag{3.35}$$

The Bellman optimality equation for q_* is:

$$
\begin{aligned}
q_*(s,a) &= \mathbb{E}\left[R_{k+1} + \gamma \max_{a'} q_*(S_{k+1}, a') \mid S_k = s, A_k = a\right] \\
&= \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma \max_{a'} q_*(s', a')\right]
\end{aligned}
\tag{3.36}
$$

In tasks with small, finite state sets, it is possible to form approximations using arrays or tables with one entry for each state (or state-action pair). This we call the tabular case, and the corresponding methods we call tabular methods. In many cases of practical interest, however, there are far more states than could possibly be entries in a table. In these cases the functions must be approximated, using some sort of more compact parameterized function representation.

3.4.2 Deep Q-Learning

As stated in the previous section, reinforcement learning focuses on solving a problem of learning how to interact with an environment by interacting with it. With the popularity of deep learning algorithms, deep reinforcement learning (DRL) presents a great success in solving highly challenging problems. In DRL, the value or policy functions are often represented as deep neural networks and the related deep learning techniques can be readily applied ([25]). In this work, we use deep Q-network (DQN), first introduced in [26]. The algorithm was developed by enhancing a classic Reinforcement Learning algorithm called Q-Learning with deep neural networks and a technique called experience replay. The algorithm was able to learn to play a wide range of Atari games and even beat the humans in most of them.

Despite its great empirical success, there exists a substantial gap between the theory and practice of DRL. The first attempt to theoretically understand DQN is presented in [25].

Considering the problem we want to solve in this work, the agent is the decision maker that needs to choose what level of NPIs (mainly lockdowns) it wants to imply. The agent interacts with an environment, in this case the SEIHRD model, in a sequence of actions, observations and rewards. At each instant k, the agent selects an action A_k from the set of actions $\mathcal{A}$. The action is passed to the SEIHRD model and modify its state (i.e. compartmental groups). The new state S_{k+1} and a reward R_k is received by the agent. Therefore, sequences of actions, observations and rewards are input to the algorithm, which then learns strategies depending upon these sequences.

The goal of the agent is to interact with the environment selecting actions in a way that maximizes future rewards. The basic idea behind many reinforcement learning algorithms is to estimate the action-value function by using the Bellman

equation as an iterative update, converging to the optimal action-value function. In practice, this approach is impractical, since the action-value function is estimated separately for each sequence, without any generalization. Therefore, it is common to use a function approximator to estimate the action-value function, like a linear function approximator or even a nonlinear function approximator such as a neural network.

In DQN algorithm, a deep neural network $Q_\theta : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is used to approximate q_*, where θ is a parameter. Two tricks are pivotal for the empirical success of DQN. First, DQN use the trick of experience replay. At each instant k, the transition (S_k, A_k, R_k, S_{k+1}) is stored into the replay memory $\mathcal{M}$ and then sample a minitbatch of independent samples from $\mathcal{M}$ to train the neural network via stochastic gradient descent. Since the trajectory of MDP has strong temporal correlation, the goal of experience replay is to obtain uncorrelated samples, which yields gradient estimation for the stochastic optimization problem. The second trick is to use a target network Q_{θ^*} with parameter θ^* (current estimate of parameter).

The algorithm used in this work is shown in Algorithm 2 ([26], [25]).

Algorithm 2: Deep Q-Network ([25])

Input: MDP $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, replay memory $\mathcal{M}$, number of iterations K, minibatch size n, exploration probability $\varepsilon \in (0,1)$, a family of deep Q-networks $Q_\theta : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, an integer K_{target} for updating the target network, and a sequence of stepsizes $\{\alpha_k\}$, $k \geq 0$;

Initialize replay memory $\mathcal{M}$ to be empty;

Initialize the Q-network with random weights θ;

Initialize the weights of the target network with $\theta^* = \theta$;

Initialize the initial state S_0;

for $k = 1 : K$ **do**

1) With probability ε, choose A_k uniformly at random from $\mathcal{A}$, and with probability $(1-\varepsilon)$, choose A_k such that $Q_\theta(S_k, A_k) = \max_{a\in\mathcal{A}} Q_\theta(S_k, a)$;

2) Execute A_k and observe reward R_k and next state S_{k+1};

3) Store transition (S_k, A_k, R_k, S_{k+1}) in $\mathcal{M}$;

4) Experience replay: Sample random minibatch of transitions $\{(s_i, a_i, r_i, s_i')\}_{i\in[n]}\}$ from $\mathcal{M}$;

5) For each $i \in [n]$, compute the target $Y_i = r_i + \gamma \max_{a\in\mathcal{A}} Q_{\theta^*}(s_i', a)$;

6) Update the Q-network: Perform a gradient descent step

$$\theta \leftarrow \theta - \alpha_k \frac{1}{n} \sum_{i\in[n]} \left[Y_i - Q_\theta(s_i, a_i)\right] \nabla_\theta Q_\theta(s_i, a_i)$$

7) Update the target network: Update $\theta^* \leftarrow \theta$ every K_{target} steps;

end

Define policy π^* as the greedy policy with respect to Q_θ;

Output: Action-value function Q_θ and policy π^*.

3.5 Omniscient Control

Finally, we present a benchmark globally optimal control policy, that is hypothetical, since it assumes that all data for the whole control horizon is known. In this case, of course, it is possible to calculate the open-loop globally optimal control for any given performance index, and we will refer to this as the omniscient control, since the control design can observe all data, without any errors or estimates. The main objective in presenting and calculating this strategy is to have a baseline for comparison of the other practically implementable strategies.

Given a performance index, Omniscient Control follows the classical recipe of optimal control. At instant $k = 1$ we calculate all u_k for k lying in the interval $(1, K)$.

The omniscient optimal control problem is described below in equations (3.37) -

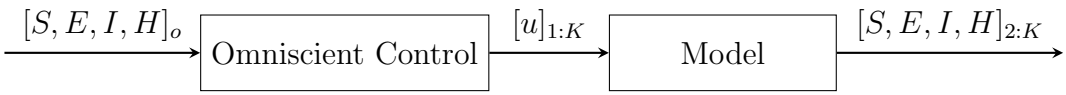

Figure 3.5: N-Steps Ahead Control Diagram.

(3.42):

$$\min \quad J = \sum_{k=1}^{K} u_k \tag{3.37}$$

$$\text{subject to:} \quad S_{k+1} = S_k - (1 - u_k)(\alpha S_k E_k + \beta S_k I_k) \tag{3.38}$$

$$E_{k+1} = E_k + (1 - u_k)(\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1 - p_1))E_k \tag{3.39}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1 - p_2))I_k \tag{3.40}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1 - p_3))H_k \tag{3.41}$$

$$H_k \leq H_{max} \tag{3.42}$$

$$0 \leq u_k \leq 1 \tag{3.43}$$

After all control variable values are calculated, we can use them in a real world environment. If the SEIHRD model parameters are exactly equal to the real world parameters, the Omniscient Control would have a perfect performance. Of course, given the multiple uncertainties and assumptions made in the model, this will almost never happen. Thus, for each instant k, the respective optimal control u_k calculated from the model is applied to the real system which responds with the actual state variables of the instant $k + 1$.

Chapter 4

Simulations

In the previous chapter, we explained the theory behind the strategies we chose to use in this work. In this chapter we put the theory into practice. The simulation consists in two cases: In the first one, the environment parameters are constant and equal to the model used to calculate the control values. Next, parameter uncertainty is introduced in the SEIHRD model and, consequently, their values will differ from the ones used in the model.

In section 4.3, we analyze how the NSAOC strategy behaves. We study the influence of parameter N on the control calculation and the consequences on each compartment of the population. In order to solve possible problems occasioned by the main objective function that minimizes the total effort, we study the impact of using different performance indexes.

In the subsequent sections, we simulate the remaining strategies proposed to compare with NSAOC strategy. First, in order to understand what might happen when no control measure is applied on the population, section 4.1 presents the results of this simulation. Section 4.2 presents constant interventions for the entire horizon in order to understand which levels of control input would be needed. Section 4.4 presents the benchmark hypothetical omniscient control strategy. The PID-Like controller proposed in [1] is presented in section 4.5. Lastly, the reinforcement learning strategy is used in section 4.6.

4.1 SEIHRD Model without intervention

In this section, no control measure is applied on the population. This is equivalent of using equations 2.1 - 2.6 to calculate the new values of each state of SEIHRD model for each instant k (i.e. each day).

The results of this simulation are shown in Figure 4.1 for k assuming values from 1 to 250. If no action is done to control the epidemic, approximately 82% of the entire population would be infected by the virus. Also, at minimum 4.59% of the

population would die. However, this number probably would be higher, since the number of hospitalized people would reach 5%, which is much more than the limit (0.8%) recommended by the World Health Organization (WHO).

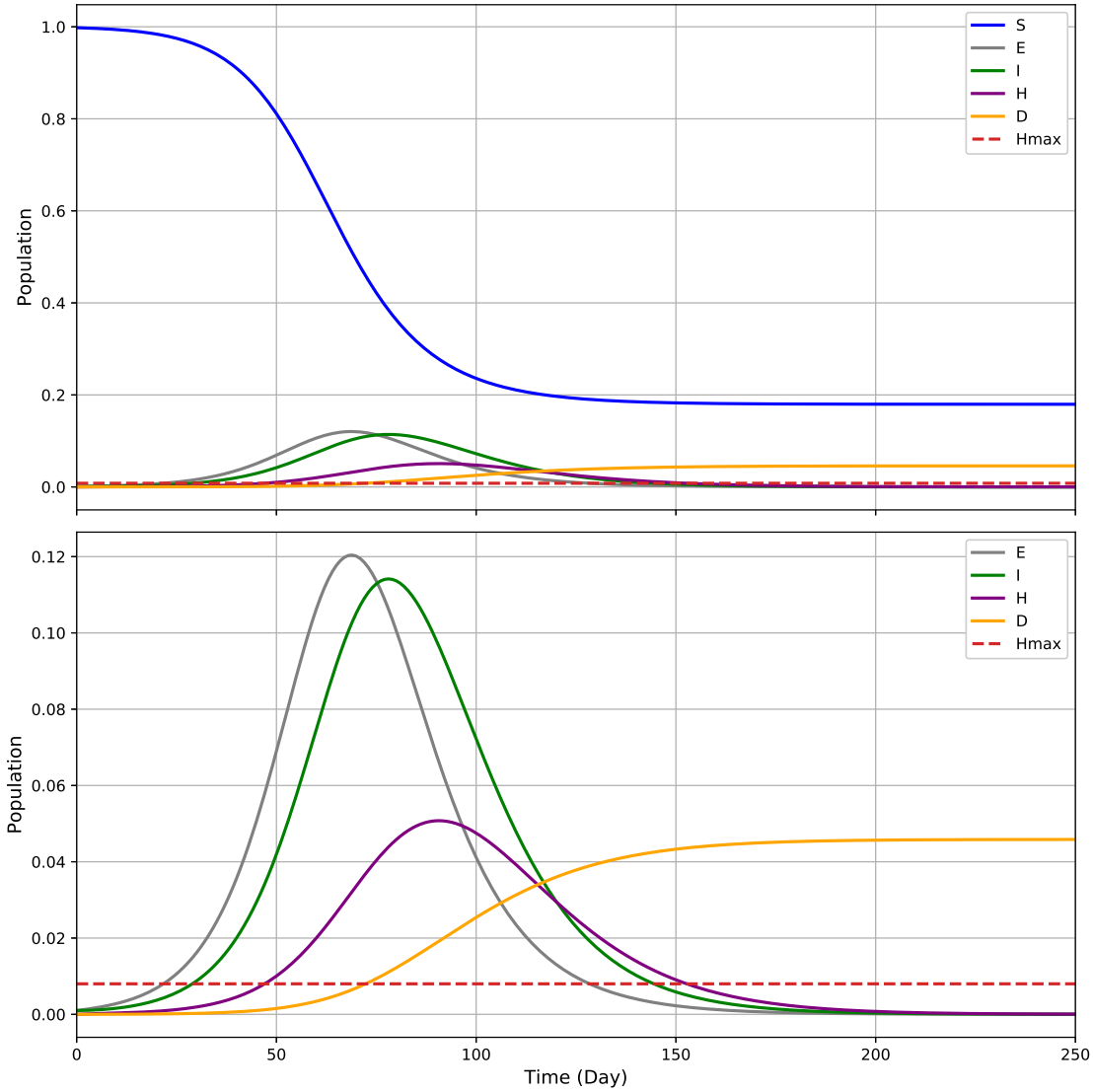

Figure 4.1: Evolution of SEIHRD model states with constant parameters and no interventions applied in the environment.

One important instant of the graph is the peak the Exposed curve. This is the moment where the epidemic starts to decline, since the number of susceptible individuals is not high enough to keep increasing the number of exposed and infected individuals. This occurs at instant $k = 70$ and the percentage of susceptible and exposed individuals are 12.03% and 50.77%, respectively.

4.2 Intervention using constant control variable

In this section, we simulate how the SEIHRD model reacts when using constant values as input control. In order to do that, we use equations (3.11) - (3.16) and replace u_k by a constant value. As stated in the previous chapter, u_k is in the interval $[0, 1]$. For this simulation, we use 6 different values:

$$U = \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}, u_k \in U \tag{4.1}$$

The parameters used in the SEIHRD model are specified in table 2.2. Figure 4.2 shows the results for k assuming values from 1 to 800. The output of this simulation is very useful to understand the impacts of using different control values.

The first observation is that when u_k is increased, the number of deaths is smaller. Also, the peaks of exposed, infected and hospitalized individuals are smaller and occur later. As a consequence, the pandemic lasts longer, since less individuals are contaminated each day and there is always a relevant number of susceptible individuals.

For certain levels of control input, the number of hospitalized people does not exceed the desired limit H_{max}. For example, the maximum number of hospitalized people with $u_k = 0.4$ is 0.775%. However, for very high control levels, the hospitalization level is kept very low, which is an indication that some NPIs can be relaxed without impacting the economy severely.

One can also note that varying u_k has the effect of changing the observed α and β parameters of the epidemic, since it affects directly the contact between susceptible, exposed and infected people.

4.3 N-Steps Ahead Optimal Control Algorithm

This section applies the NSAOC algorithm discussed in chapter 3. First, in order to show how the algorithm performs, a simulation using a SEIHRD model with constant parameters is used. The parameters of the SEIHRD model are resultant from an estimation study given the data available from the observations and also some additional considerations ([1]). In the real world, we know that these values are subjected to disturbances and their real values are very likely to change, for instance because we already have many different COVID-19 variants.

The algorithm assumes $N = 10$ and uses only one objective function, which is to only minimize the total control input over the time horizon (equation(4.2)). In the subsequent sections, an analysis for these two statements is presented.

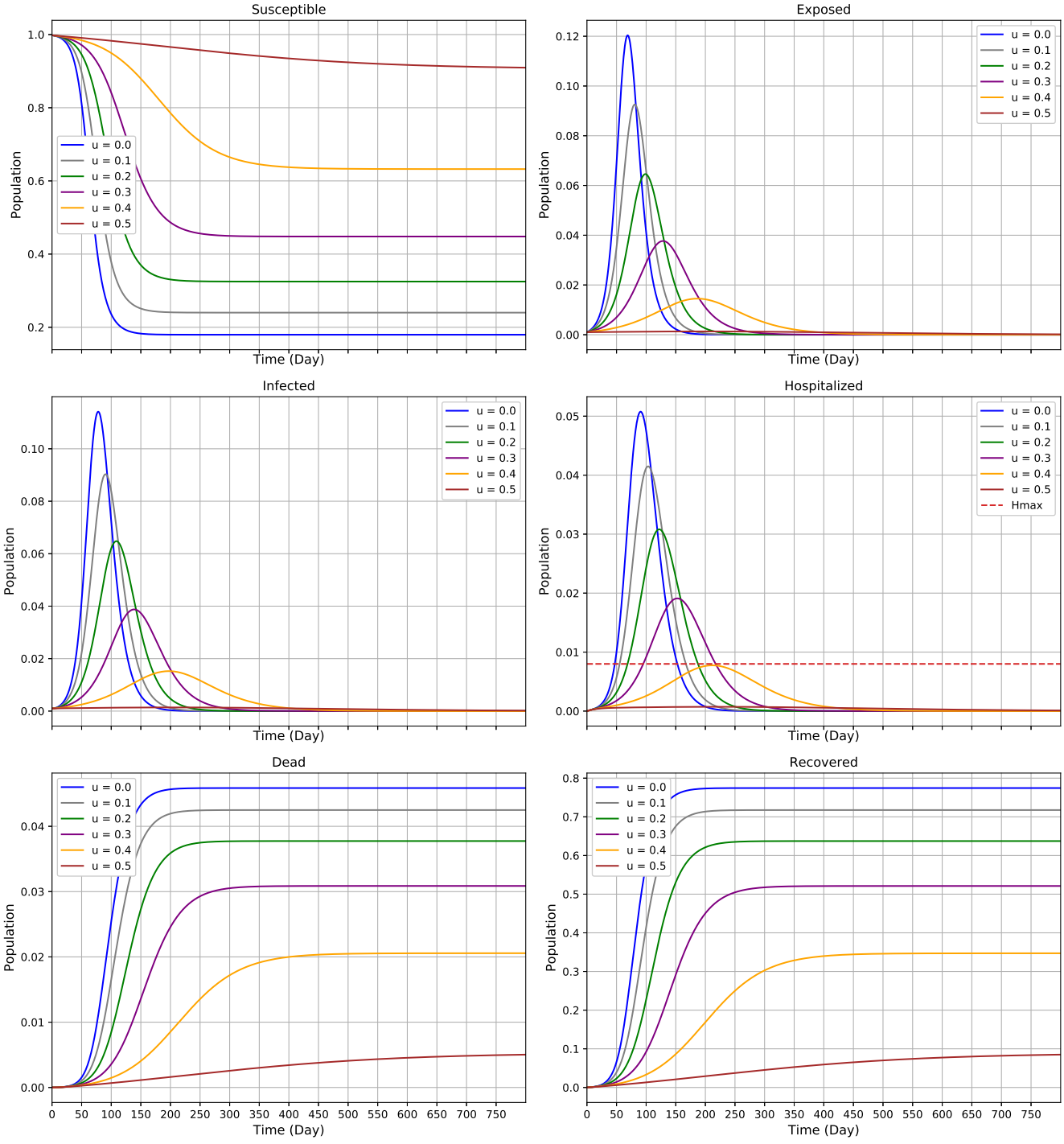

Figure 4.2: Results of a SEIHRD model with constant parameters using different levels of constant control input.

$$J_1 = \sum_{k}^{k+N} u_k \tag{4.2}$$

For the first simulation, we use the values of table 2.2 and also consider that they do not change over time. In terms of Figure 3.2, we suppose that the model used to calculate the control variables (inside block NSAOC) is the same as the model used in the environment. The simulation horizon is $K = 600$ days and we use Ipopt optimizer (as in any other simulation containing optimization in this work).

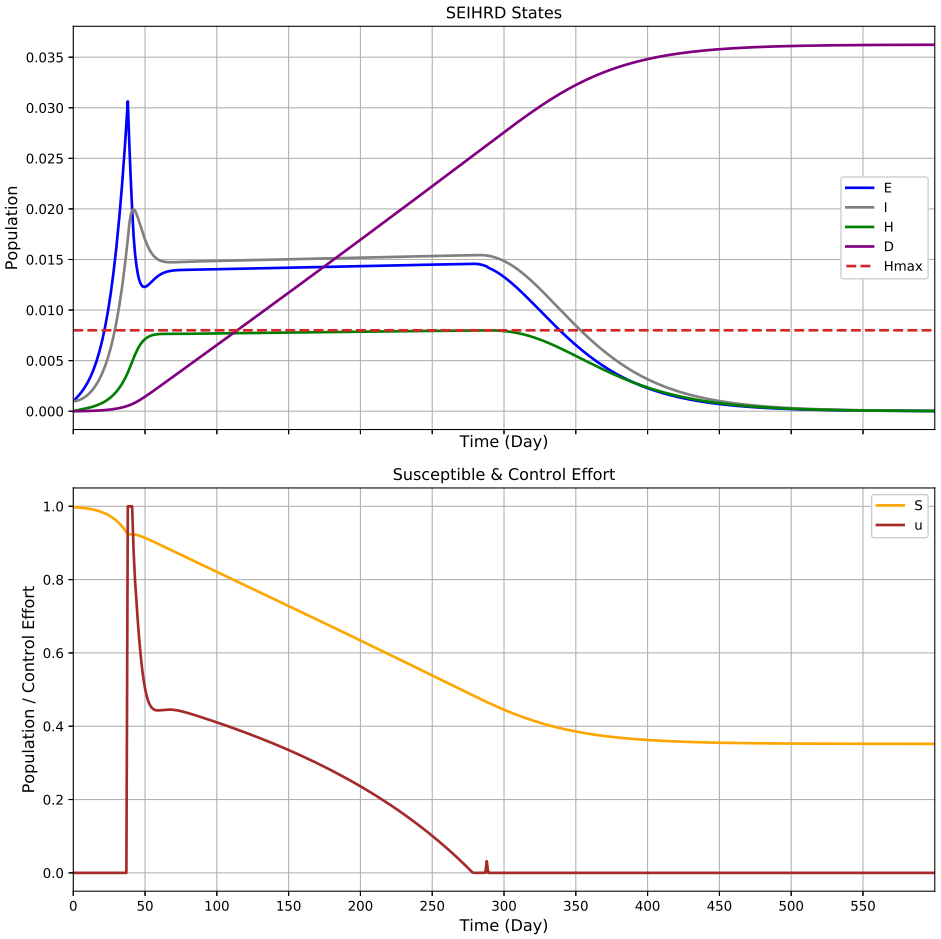

Figure 4.3: Results of a SEIHRD model with constant parameters using NSAOC algorithm with $N = 10$, time horizon $K = 600$ and using performance index J_1.

In NSAOC strategy, the only parameter we need to adjust is the parameter N and, indeed, this is one of its good features. As stated earlier, this parameter represents the number of steps ahead of the actual instant the calculation of the optimal solution is carried out. For example, if we use $N = 10$ (Figure 4.3), the epidemic can be controlled and the number of hospitalized people never exceeds the specified limit of 0.8%. Also, the control input is only greater than zero when the number of hospitalized people reaches a dangerous level. Then, a severe lockdown

Parameter	Distribution
α	Normal(0.1786, 0.05)
β	$\alpha/2$
γ^{-1}	Normal(5.1, 2)
ζ^{-1}	Normal(14.7, 1)
δ^{-1}	Normal(5.5, 2)
η^{-1}	Normal(11.2, 4)
ϵ^{-1}	Normal(16, 4)
μ^{-1}	Normal(14, 1)
p_1	Normal(50%, 20%)
p_2	Normal(19%, 10%)
p_3	Normal(15%, 3%)

Table 4.1: Parameter values for SEIHRD model on a simulation with parameter uncertainty.

is put in place ($u = 1$) for a week, approximately. After the contamination starts to decrease, the input control can be relaxed to a lower level.

The final percentage of deaths is 3.62%, which is still a high percentage, but lower than the first simulation where no control measure is applied. Also, the peak of infected and exposed individuals are smaller, since the control applied reduces the contact between susceptible and individuals carrying the virus.

In the real world, only expected values of most of the parameters of the SEIHRD model are available. In the next simulation, we consider a SEIHRD model with parameter uncertainty. As shown in table 4.1, in this work we use normal distribution to represent the parameter uncertainty, as it is one most used distributions and fits well in the model. Consequently, multiple scenarios could occur. Thus the simulation is repeated 1000 times in order increase the possibility of considering adverse scenarios in the analysis.

The results are presented in Figure 4.4. The lines represent the mean of all states for each instant, while the dashed areas represent the variation that was identified in the simulations. The strategy applied still worked well, obeying the value limit specified for hospitalized people, even in the worst case scenario where the maximum hospitalization level was 0.8%. This is the main advantage of using NSAOC and it is only possible due to the feedback present in the algorithm, where every day the actual measurements of each state of the model are used to calculate a new control value.

In order to compare with the previous simulation where the parameters were constant, three scenarios are used: The worst case scenario, the best case scenario and the mean of all simulations. Table 4.2 summarizes the comparison.

Considering the mean case, the peak of infected population is higher than the

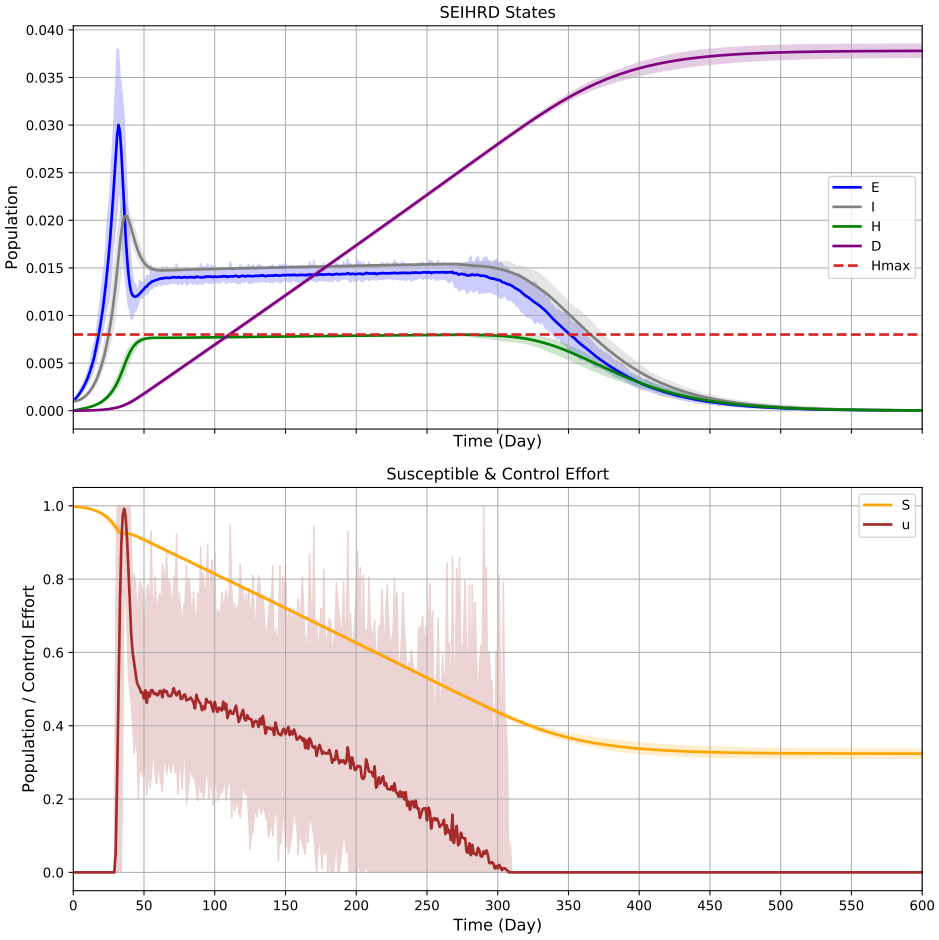

Figure 4.4: Results of a SEIHRD model with parameter uncertainty using NSAOC algorithm with $N = 10$, time horizon $K = 600$ and using performance index J_1.

constant case, as the algorithm cannot predict the rate of spread of the virus. The direct consequence is seen in the total control applied, much higher in the mean case (90.26) due to the need to increase the control input and thus avoid contact of susceptible individuals with infected and exposed individuals. This also leads to a higher total number of deaths, as the epidemic is better stabilized in the Constant case.

If we change the performance index and choose one that only minimizes the number of daily deaths, the result would be the following function:

$$J_D = \sum_{i=k}^{k+N-1} \sum_{j=k+1}^{k+N} D_i - D_j \tag{4.3}$$

The results are shown in Figure 4.5. The resultant control would assume the maximum level ($u_k = 1$) for a certain period, until the number of infected and exposed individuals go to 0. Then, the virus is eliminated from the population.

SEIHRD Parameters	Max I	Max E	Max H	Deaths	Sum u_k
Constant	1.997%	3.063%	0.8%	3.624%	74.82
Var. - Worst Case	2.187%	3.798%	0.8%	3.851%	196.0
Var. - Best Case	1.946%	2.307%	0.796%	3.715%	31.36
Var. - Mean Case	2.049%	3.002%	0.799%	3.781%	90.26

Table 4.2: Result comparison of four different cases using strategy NSAOC with $N = 10$ and using performance index J_1.

However, this is a hypothetical case considering that it is possible to interrupt every contact between individuals, which is not an option in the real world.

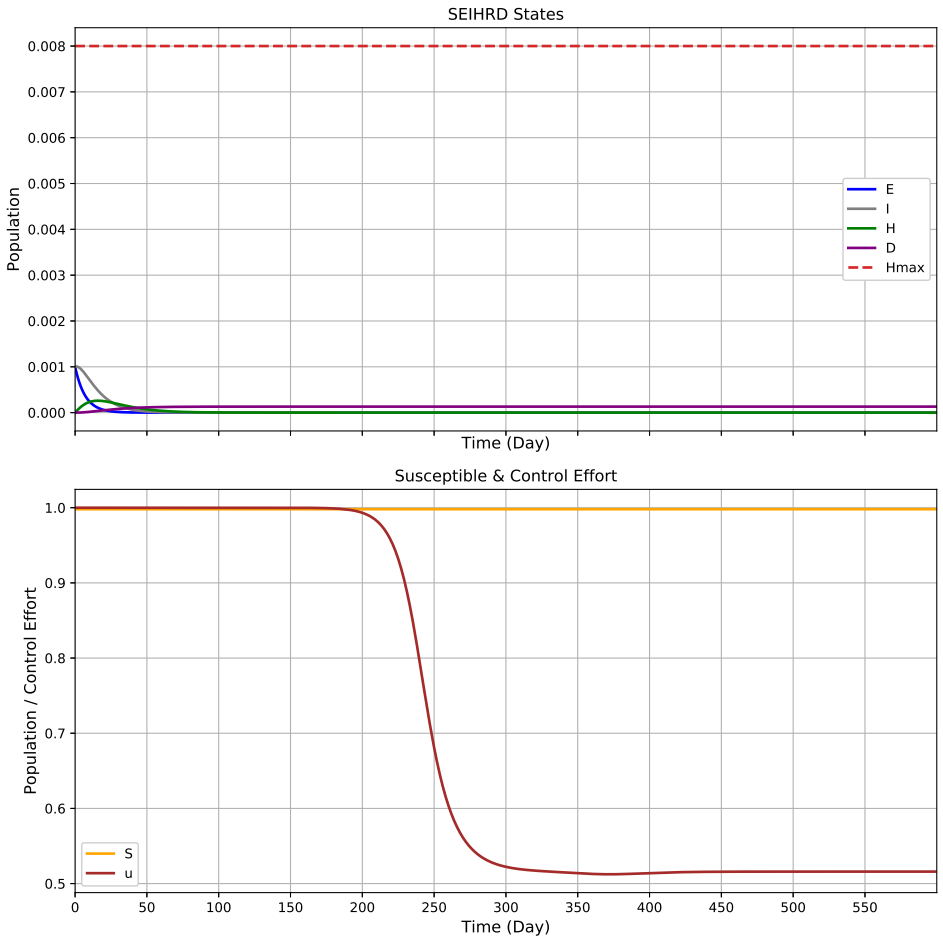

Figure 4.5: Results of a SEIHRD model using NSAOC algorithm with $N = 10$ and time horizon $K = 600$ using performance index J_D.

4.3.1 Objective function comparison

In the previous section, only one objective function was used, with the main goal being to minimize the total control input while keeping the total number of hospi-

talizations below a desired level. What if we use different performance indices? This section compares three different functions to analyze the benefits of each one.

The performance indices used in this section are the following:

$$J_1 = \sum_{k}^{k+N} u_k \tag{4.4}$$

$$J_2 = \sum_{k}^{k+N} wu_k \tag{4.5}$$

$$J_3 = \sum_{k}^{k+N} wu_k + \sum_{i=k}^{k+N-1} \sum_{j=k+1}^{k+N} (u_i - u_j)^2 \tag{4.6}$$

The first objective function (4.4) is the same we used in the previous sections and the main goal is to simply minimize the total control input applied on the population. This performance index is used in [1] and we also use in this work in order to compare the strategies.

The second objective function (4.5) is similar to the first one, but it has an extra vector w. This is a weight vector and we define it as $w = [N^2, (N-1)^2, (N-2)^2, ..., 1^2]$. This modification results in giving more importance to closer instants, making the corresponding control input higher than more distant instants.

The third objective function (4.6) adds a slew rate penalty to the second one. As we will observe in the simulations, the second objective makes the controller vary a lot between high values and low values. The slew rate term penalizes big jumps in the control, smoothening the control signal, which is desirable.

The first simulation uses a SEIHRD model with constant parameters. The results of the simulation using different objective functions is shown in Figure 4.6 and table 4.3.

Although the objective function J_2 has the lowest total control input applied, the difference between the inputs on each day varies a lot, which can be impossible to implement in a real situation. An interesting fact is that, even though the peak of state I is higher when using J_3, the peak of state E is higher when using J_2. This is explained by the slew rate penalty (also parameter N has an influence, as we will observe in next section) added only in J_3. The controller identifies that the hospitalization level can exceed the limit and acts accordingly. However, due to the mentioned penalty, use of index J_3 smoothens the control decrease, leading to a higher peak of infected individuals, unlike J_2 which results in a sudden control decrease.

The objective function J_3 has similar behavior to J_1, but results in a smoother control. Furthermore, the hospitalization level is always below the desired level,

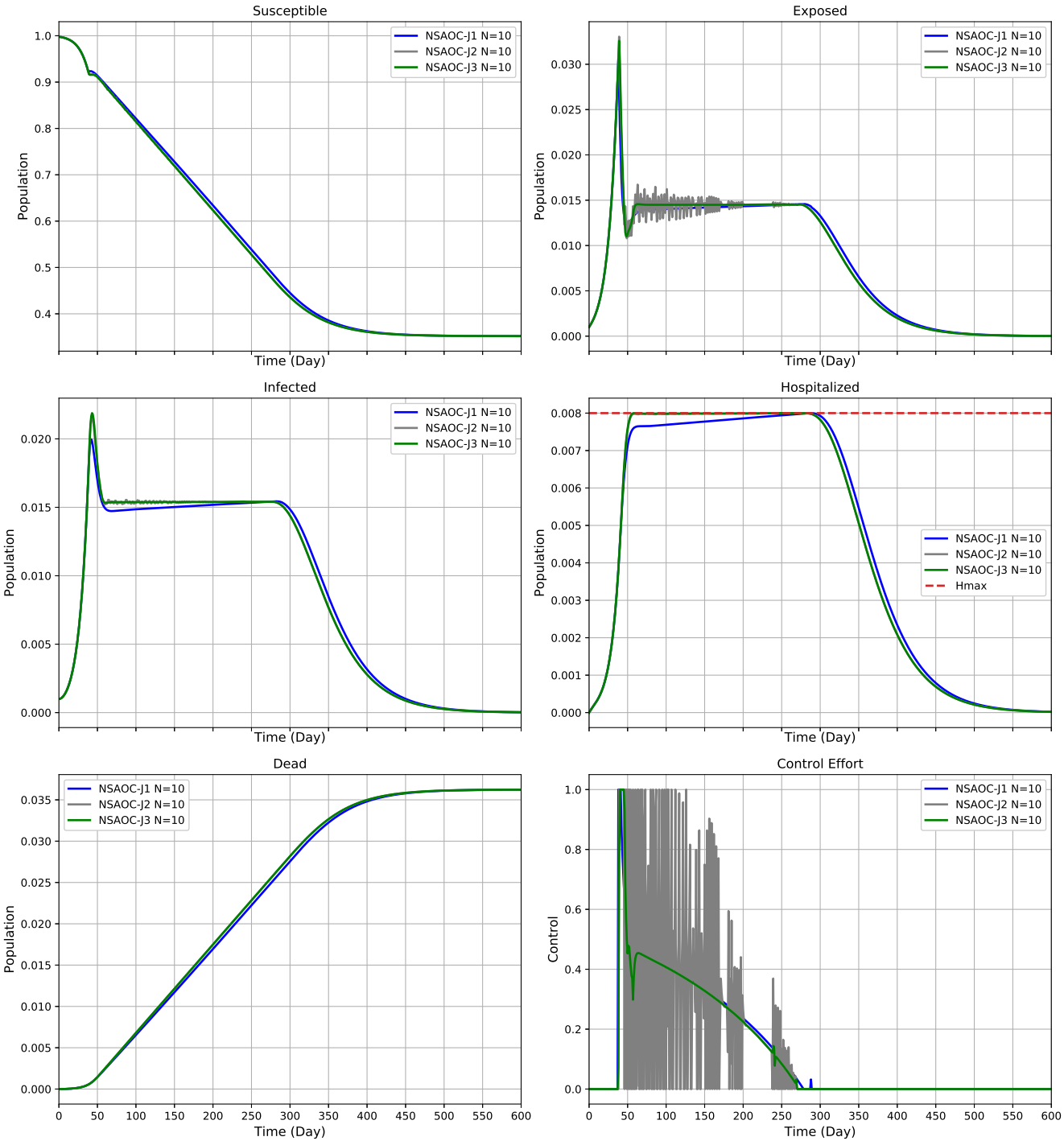

Figure 4.6: Comparison of each state from SEIHRD model and the control input, using NSAOC strategy ($N = 10$) and performance indices J_1, J_2 and J_3 in a scenario with constant parameters.

with J_1 having a higher slack compared to H_{max}, but also having a higher total control input.

Obj. Function	Max I	Max E	Max H	Deaths	Sum u_k
J1	1.997%	3.063%	0.8%	3.624%	74.82
J2	2.16%	3.304%	0.8%	3.624%	70.91
J3	2.188%	3.258%	0.8%	3.625%	72.42

Table 4.3: Comparison between different objective functions using SEIHRD with constant parameters.

4.3.2 Parameter N: Impact Study

In this section, we vary the parameter N of the NSAOC algorithm and experiment in the SEIHRD model without parameter uncertainty. We consider all three objective functions discussed in the last section 4.3.1, defined earlier in equations (4.4), (4.5) and (4.6).

As stated before in this work, COVID-19 is a disease with a relatively high incubation period and the consequences of the actions taken today will most likely only appear in a week or even more. Therefore, parameter N must be chosen wisely in order to predict specially the hospitalization level early enough.

First, we investigate the impact of different values of N on the hospitalization level and number of deaths. As shown in Figure 4.7, when using objective function J_1 the minimum value we should use for N is 9, while for objective functions J_2 and J_3 this value changes to 10. This strategy maintains the hospitalization level below the desired limit H_{max}.

We also note that for values of N below 9 (using J_1) and 10 (using J_2 and J_3), the optimization problem is infeasible. This occurs because it cannot see enough steps ahead to identify that the people exposed and infected at the actual instant will become ill a few days later and the number of hospitalized people will exceed the limit H_{max}, violating the constraint (and making $N < 9$ step ahead problem infeasible).

As long as N increases (and is larger than 10), the maximum hospitalization level stabilize near the specified limit H_{max} for all performance indexes, even though it takes more time to reach the limit using J_1, as shown in previous simulations. The number of deaths seems to oscillate near the value 3.8%, indicating that changing parameter N does not result in better results. So, considering only 4.7, we conclude that the best choice for N is 10, since it results in faster calculations.

When it comes to total control effort, a different behavior when using different objective functions can be also noted. Again, for N smaller than 10, the algorithm

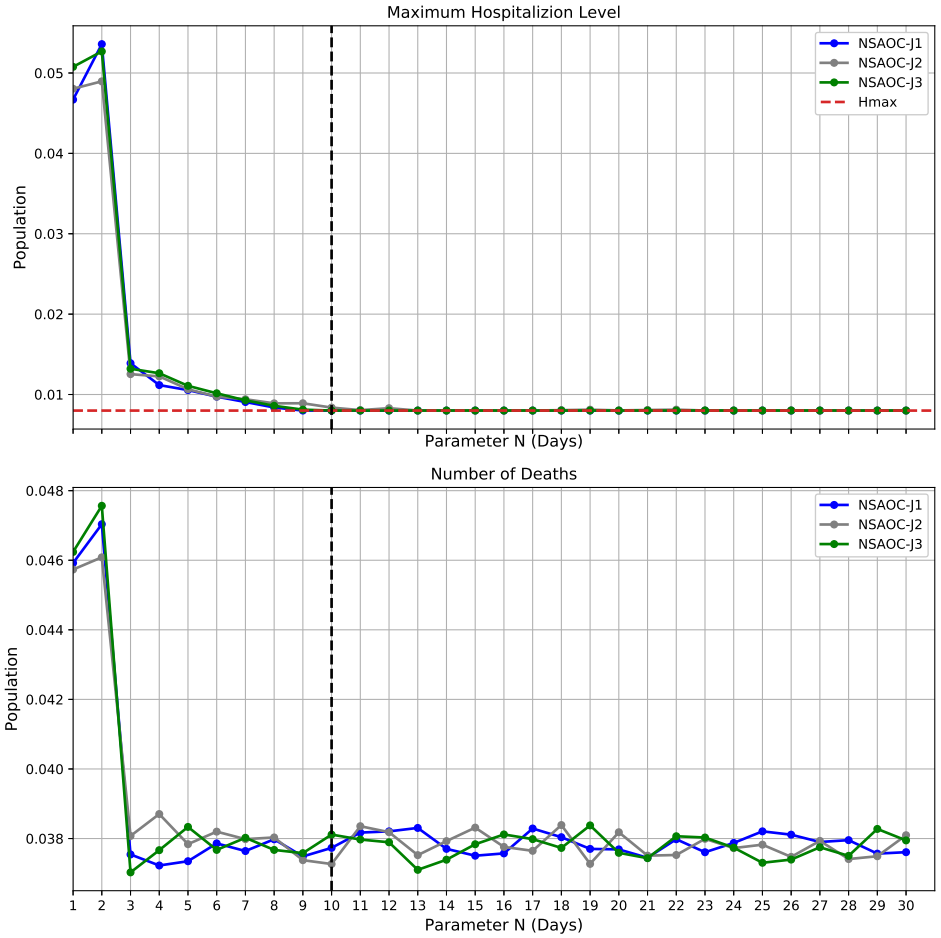

Figure 4.7: Maximum value of hospitalization level and final number of deaths using NSAOC strategy with different values of parameter N and objective functions J_1, J_2 and J_3 in a SEIHRD model with constant parameters.

is not able to find a feasible solution. For higher values of N, the total control effort increases as N also increases when using objective function J_1. However, when using functions J_2 and J_3, the total control does not vary much as N increases, because the controller only acts when it is really needed.

In Figure 4.9, the similarity of behavior between NSAOC-J2 and NSAOC-J3 is even more evident, especially when parameter N is relatively high (greater than 15). The peak number of infected individuals is the same in both strategies and also the peaks of exposed individuals are very similar. When using NSAOC-J1, the peaks are smaller as N increases, until $N = 18$, when the peaks tend to not be affected by any increase of N anymore.

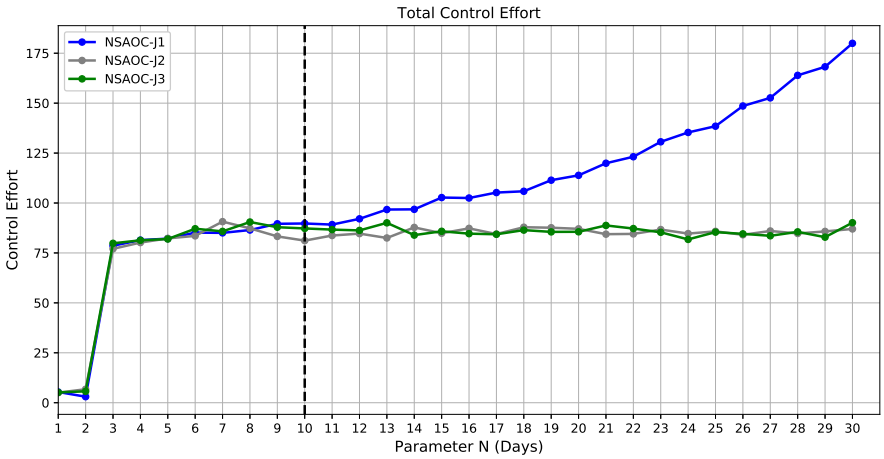

Figure 4.8: Total control input using NSAOC strategy with different values of parameter N and objective functions J_1, J_2 and J_3 in a SEIHRD model with constant parameters.

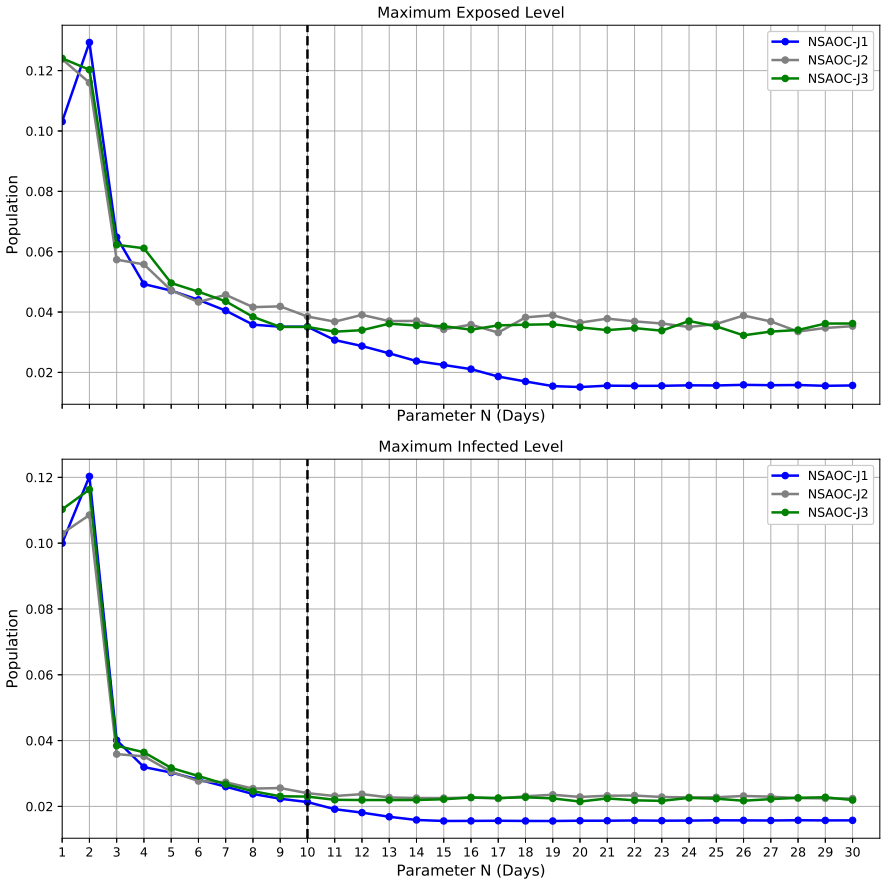

Figure 4.9: Maximum value of exposed and infected individuals using NSAOC strategy with different values of parameter N and objective functions J_1, J_2 and J_3 in a SEIHRD model with constant parameters.

4.4 Omniscient Control Algorithm

In this section, we investigate the response of the SEIHRD Model to omniscient optimal control. In the first simulation, all SEIHRD parameters are constant based on table 2.2. After, parameter uncertainty is considered in the SEIHRD model according to table 4.1.

When we apply the control inputs to the SEIHRD model without parameter uncertainty, it is expected that the environment reacts exactly as expected. In that case, the control strategy succeeds.

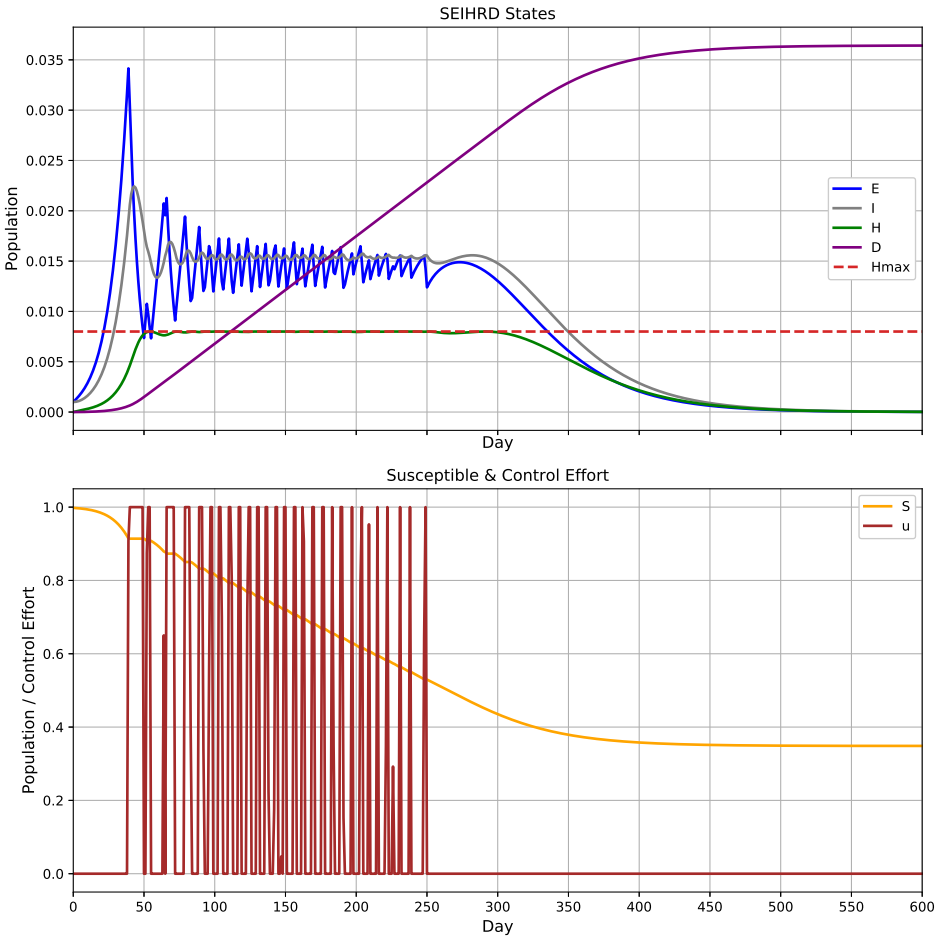

Figure 4.10: Results of a SEIHRD model with constant parameters using omniscient strategy and time horizon $K = 600$.

The results are shown in Figure 4.10. At the beginning of the period simulated, no restrictions are applied and the epidemic grows exponentially as the hospitals still have capacity to treat all sick people. When the number of individuals that requires hospitals starts to increase, the first restrictions are applied. After that, periods of full lockdown and no restrictions at all are alternated. Due to this behavior, the

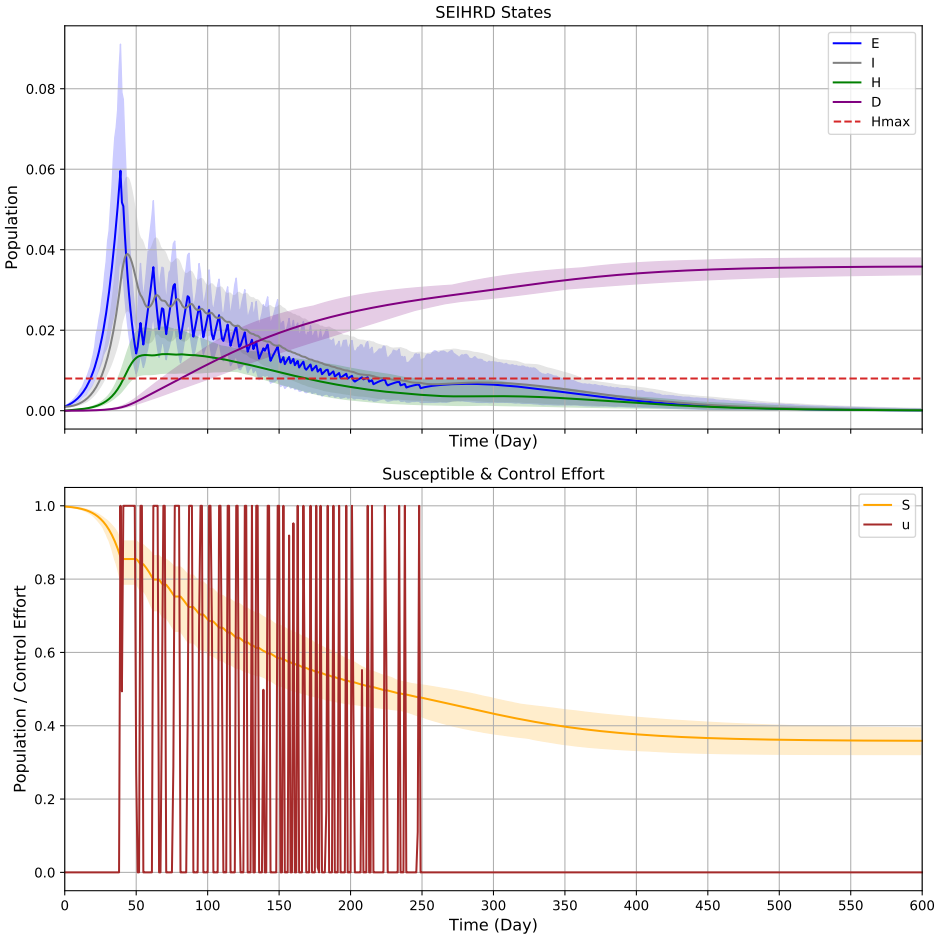

Figure 4.11: Results of a SEIHRD model with parameter uncertainty using omniscient strategy and time horizon $K = 600$.

number of exposed and infected individuals keeps oscillating the entire period. In order to avoid that behaviour, a different performance index could be used, like $J = u_k^2$. The results would present a smoother control curve with the cost of increasing the total control input.

In the end of the time horizon studied, 34.84% of the population are still in the susceptible group, 3.64% died and the total control input was 70.82. The number of hospitalized did not exceed the limit H_{max} (0.8%).

In the next simulation, we consider parameter uncertainty in the SEIHRD model in accordance with table 4.1 and the same omniscient control strategy is applied.

In order to include all scenarios in the analysis, the simulation is repeated 1000 times. The mean of state values over all simulations are shown in Figure 4.11 and the dashed area represents its variation.

The results show that the number of hospitalized individuals exceeds its limit between days 43 and 44 and only returns to an acceptable level at instant 170. Since we do not have any kind of feedback on the state of each population compartment, it

SEIHRD Parameters	S	Max(I)	Max(H)	Deaths	Sum(u)
Constant	2.243%	3.415%	0.8%	3.643%	70.82
Variable	3.895%	5.963%	1.407%	3.589%	70.82

Table 4.4: Result comparison using Omniscient Control on a SEIHRD model with constant parameters and parameter uncertainty.

is very hard to identify if the control strategy can fail or not. One good real example would be a bad estimation on the value of α and β, leading to an increase on the curve of infected and exposed people that would impact directly on the number of hospitalized people.

The results are summarized in table 4.4. We can observe that there is a peak of people needing hospital care of 1.407% (difference of 0.607% to the specified limit 0.8%) when we allow some variance on SEIHRD model parameters. The number of deaths should be much higher in this case if it is supposed that the probability of not surviving is much higher if no hospital is available.

4.5 PID-Like Control

In this section, we investigate the response of the SEIHRD Model to the PID-Like control proposed in [1] and explained in section 3.3. In the first part we consider that all parameters are constant based in table 2.2. The time horizon studied has a size of $K = 1200$ days.

We explained how Pazos et al. ([1]) developed the control calculation that resulted in equation (3.24). As in their work, we also use $k_p = 1$ to illustrate how the control strategy performs in a SEIHRD model.

The results are shown in Figure 4.12. The control is successful in keeping the number of hospitalized individuals below H_{max}. However, there is still a margin that could be used by relaxing the input control. This could be achieved by lowering the value of k_p or even including it in the optimization problem instead of using a constant value 1.

In the end of the time horizon, 3.099% of the population died and the total control input was 190.94. The number of hospitalized never exceeded the limit H_{max} (0.8%), reaching a peak of 0.538%. The peak number of states I and E were 1.297% and 1.448%, respectively, resulting in a controlled epidemic.

Compared to the previous presented strategies, PID-Like strategy was able to present less deaths in the end, although with a high cost represented by the high total control value. Also, the hospitalization level was always kept in a secure level.

In the next simulation, the PID-Like strategy is applied in a SEIHRD environ-

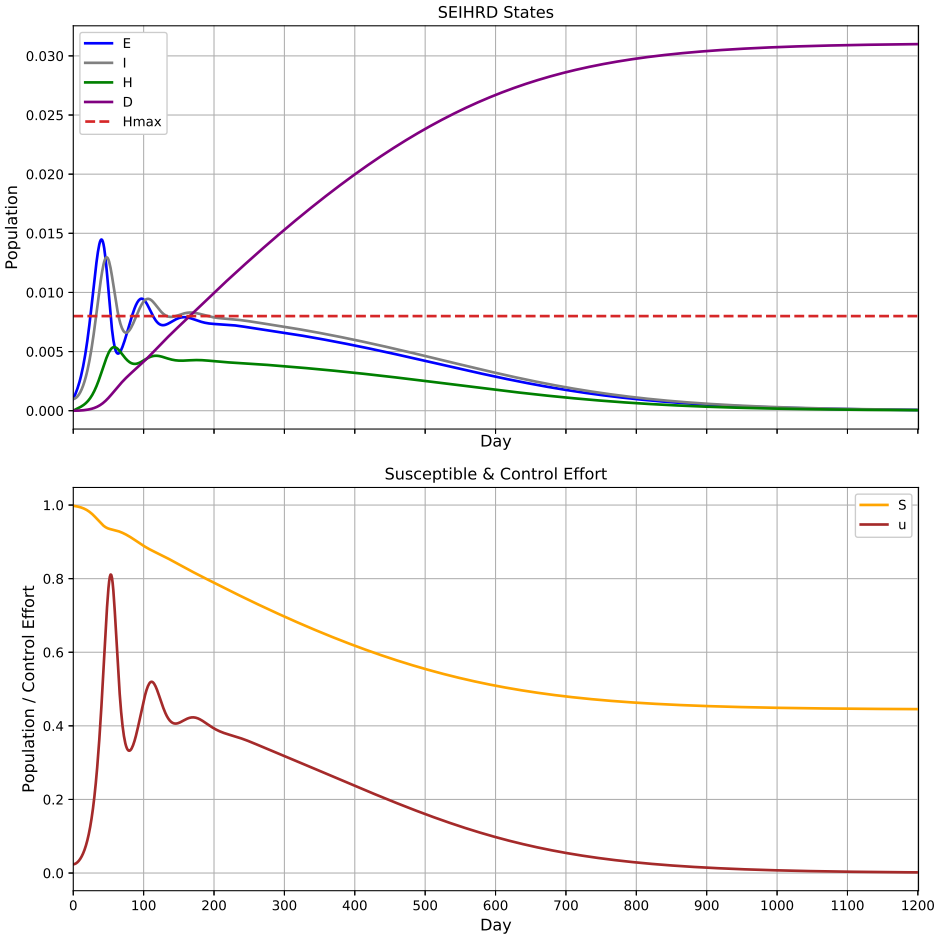

Figure 4.12: Results of a SEIHRD model with constant parameters using a PID-Like controller ([1]) and time horizon $K = 1200$.

ment with parameter uncertainty. The parameters vary according to table 4.1.

The simulation consists in running the SEIHRD model for 1000 times and extract the mean of all results and its variance.

The results are shown in figure 4.13. The peak of infected people is higher when parameter variation is allowed. However, since the PID-Like controller observes this increase, it reacts with a stronger control input, being able to maintain the number of hospitalized people below the maximum value H_{max}.

The hospitalization level is well controlled and it does not show a high variance, being always below H_{max}. The result values are shown in table 4.5.

SEIHRD Parameters	Max I	Max E	Max H	Deaths	Sum u_k
Constant	1.297%	1.448%	0.538%	3.099%	190.94
Var. - Worst Case	1.647%	2.111%	0.611%	3.358%	268.61
Var. - Best Case	1.302%	1.422%	0.538%	3.247%	168.22
Var. - Mean Case	1.457%	1.708%	0.571%	3.305%	213.24

Table 4.5: Result comparison using PID-Like Controller proposed by [1].

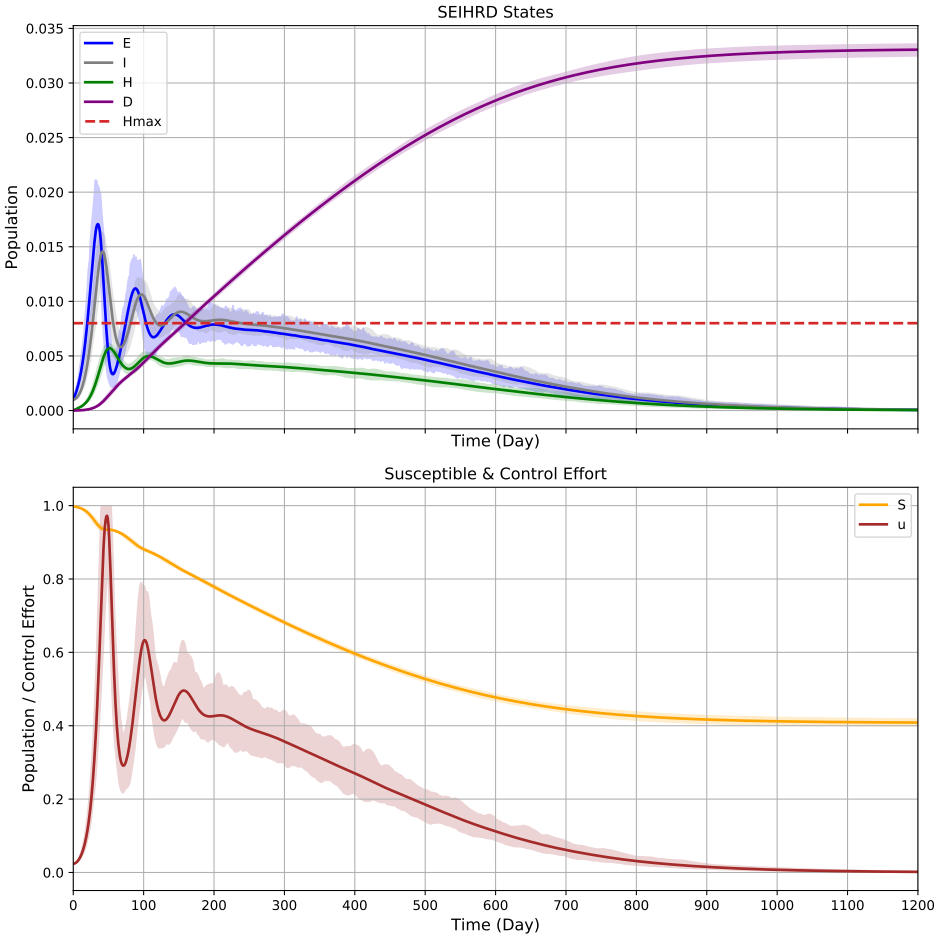

Figure 4.13: Results of a SEIHRD model with parameter uncertainty using a PID-Like controller ([1]) and time horizon $K = 1200$.

4.6 Reinforcement Learning

The last algorithm used is the DQN (Deep Q-Learning). As introduced in the last chapter, the problem to be solved can be regarded as a game that can be played multiple times by using deep reinforcement learning algorithms so the agent can

learn how to interact with the environment to gain higher rewards.

At each instant, the agent receives a representation of the current environment's state. Based on the actual state, the agent chooses one of the available actions, which is the control input that is applied in the environment. The values allowed are the following:

$$A = \{0, 0.25, 0.5, 0.75, 1\} \tag{4.7}$$

The environment responds with a new state and a immediate reward. The reward is evaluated by equation (4.8), which is based on [27]. The first part of the equation is responsible for penalizing the actions (or sequence of actions) that lead to a level of hospitalized people greater than the desired maximum. The second part tries to minimize the total control input applied in the environment. So, in general, this strategy applies the same ideas used in the optimization algorithm.

$$r_k = -\max\left(\frac{H_k - H_{max}}{H_{max}}, 0\right) - 0.1\frac{u_k^{3/2}}{4^{3/2}} \tag{4.8}$$

After training over 1000 episodes, the agent was capable of selecting the right actions in order to minimize the total control input and also not exceed the hospitalization limit level. As shown in Figure 4.14, the agent successfully understood how to control the level of hospitalized people and also tried to minimize the total effort applied. Instead of waiting to the last moment to react to the pandemic, it started to apply lighter lockdown earlier than the competing controls.

In the end, the total control input applied was 83.5 and 3.502% of the population died. The peak of hospitalized individuals was 0.788%, presenting a small margin compared to the specified limit $H_{max} = 0.8\%$.

When the same trained agent acts in a SEIRHD model with parameter uncertainty, in most cases, the strategy also succeed in controlling the epidemic, as show in Figure 4.15. In order to cover all scenarios, the same simulation was run for 1000 times. The dashed green area represents the variance of the number of hospitalized individuals and it is visible that the area exceeds the limit H_{max} in some cases.

In table 4.6, the values of the simulations are shown. The simulation of a SEIHRD with parameter uncertainty was divided in three final results: The worst case, the best case and the mean over all simulations. The agent learned very well how to react to a case with constant parameters. In an environment with parameter uncertainty, the agent could also control the epidemic, but not in all cases. The agent learned to interact with a specific model with defined parameters. When the parameters changed, the agent tried to apply the same logic learned before, but in some cases it did not as expected.

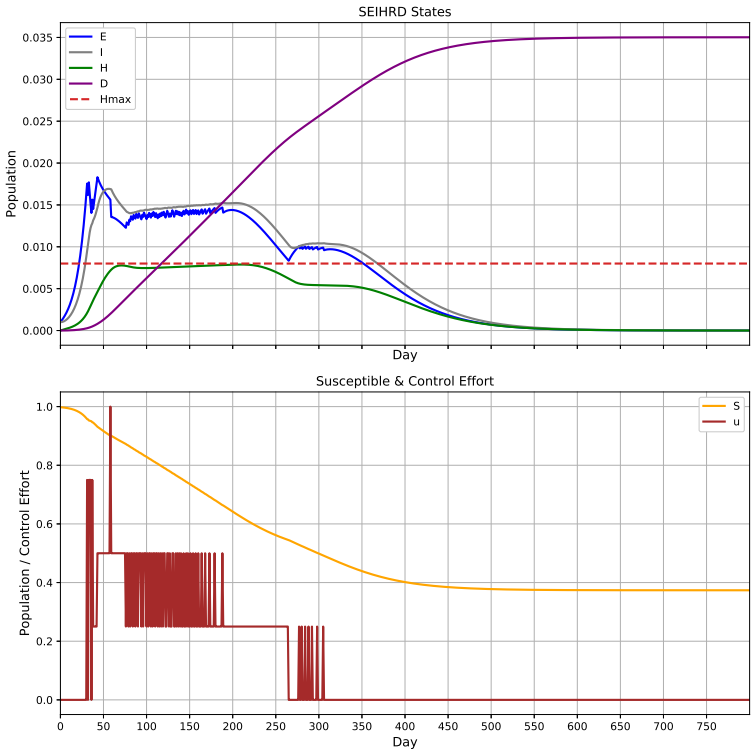

Figure 4.14: Results of a SEIHRD model with constant parameters using Reinforcement Learning Algorithm and time horizon $K = 800$.

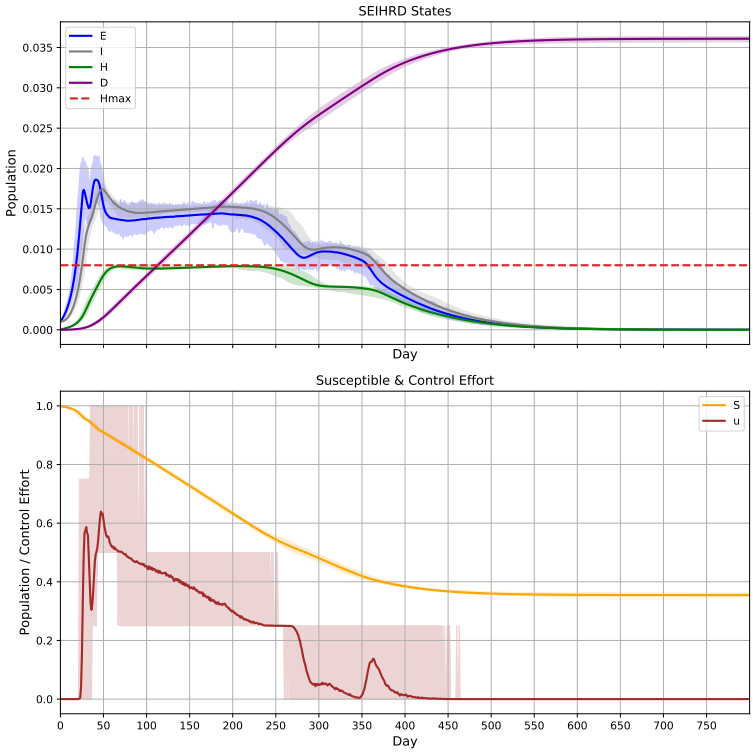

Figure 4.15: Results of a SEIHRD model with parameter uncertainty using Reinforcement Learning Algorithm and time horizon $K = 800$.

SEIHRD Parameters	Max I	Max E	Max H	Deaths	Sum u_k
Constant	1.692%	1.831%	0.788%	3.502%	83.5
Var. - Worst Case	1.816%	2.166%	0.806%	3.636%	196.25
Var. - Best Case	1.623%	1.619%	0.776%	3.57%	62.5
Var. - Mean Case	1.742%	1.862%	0.79%	3.609%	102.12

Table 4.6: Result comparison of four different cases using strategy Reinforcement Learning Algorithm.

4.7 Result Comparison

This section brings all strategies together and compares the results. When using NSAOC, the parameter N assumes the value 10. Therefore, the following strategies are used:

- NSAOC-J1 (N=10)
- NSAOC-J2 (N=10)
- NSAOC-J3 (N=10)
- Omniscient Control
- PID-Like Control (proposed in [1]).
- Reinforcement Learning

The first simulation applies the above strategies in a SEIHRD environment with constant parameters. The results are shown in Figure 4.16. The control input when using NSAOC-J2 and Omnscient strategies work approximately as a On-Off policy, while the other strategies apply smooth changes from one time instant to another. As a consequence, the Exposed, Infected population keep oscillating when using the first two strategies, specially the Omniscient strategy. The PID-Like strategy tends to choose safer decisions as it starts acting before the others and also tries to respond actively to the number of infected individuals.

The aggressiveness of the strategies affect directly the first peaks of infected and exposed people. The Omniscient strategy has the highest peaks in the infected and hospitalized states I and H, as this is the most aggressive strategy. Right after we can find NSAOC-J2 and NSAOC-J3. One interesting fact is that the peak of exposed people is higher when using NSAOC-J2 whilst the peak of infected people is higher when using NSAOC-J3. This is explained by the extra factor present in objective function $J3$ that smoothens the changes applied in the control input. So, instead of forcing the values to be almost only 0 or 1 (like $J2$), it tries to choose intermediate

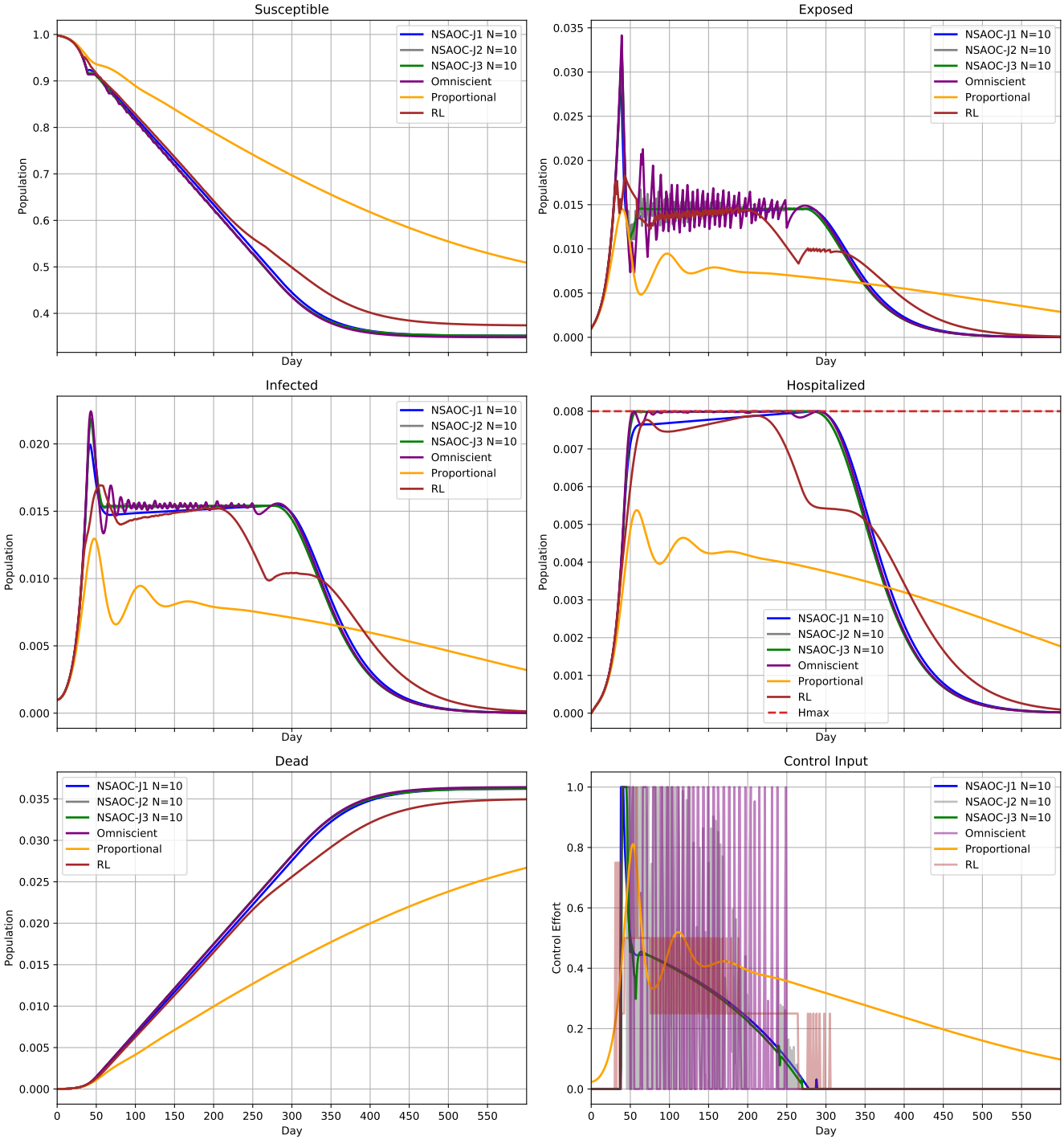

Figure 4.16: Results of a SEIHRD model with constant parameters using different strategies.

Strategy	Max(I)	Max(E)	Max(H)	Deaths	Sum(u)
NSAOC-J1	1.997%	3.063%	0.8%	3.624%	74.82
NSAOC-J2	2.16%	3.304%	0.8%	3.624%	70.91
NSAOC-J3	2.188%	3.258%	0.8%	3.625%	72.42
Omniscient	2.243%	3.415%	0.8%	3.643%	70.82
Prop.	1.297%	1.448%	0.538%	3.099%	190.94
RL	1.692%	1.831%	0.788%	3.502%	83.5

Table 4.7: Result values of a SEIHRD model with constant parameters using different strategies.

values to smooth the lockdown level descent. As a consequence, the interactions are higher after the exposed peak is reached, making the peak of infected people higher when using NSAOC-J3.

The number of deaths does not differ between all NSAOC strategies and Omniscient strategy, presenting a final value of 3.624% (3.625% using J_3 and 3.623% for Omniscient). When using RL strategy, the total control input increases and the number of deaths decreases. The same behavior occurs with Proportional strategy, what gives us an indication that the final number of deaths decreases if you increase your total control input.

The level of hospitalized people is below the limit H_{max} for all strategies. The hospitalized graph is amplified in Figure 4.17 in order to give a better view on levels closer to H_{max}. The PID-Like strategy keeps the level considerably below the lower bound. This is achieved by applying a stronger total control in the environment, as presented in table 4.7. RL and NSAOC-J1 strategies present a greater slack compared to the specified limit H_{max}. The remaining strategies (NSAOC-J2, NSAOC-J3 and Omniscient) present curves very close to the limit.

Thus, considering that SEIHRD parameters remain constant during the entire time horizon, strategies NSAOC-J2 and omniscient are impossible to implement due to their high frequency behavior. Even thought PID-Like strategy presents good levels of hospitalized and deaths, the control could be more relaxed to not impact much in the economy. Finally, strategies NSAOC-J1 and NSAOC-J3 present the most balanced results. If the estimation of the parameters of the environment is reliable, NSAOC-J3 strategy offers the best result. If not, then NSAOC-J1 is the best recommendation, as it reaches a lower level of hospitalized population with slight increase in total control applied compared to NSAOC-J3 strategy.

When parameter variation is allowed in the SEIHRD model, the results are not as good as using constant parameters, specially for strategies with more aggressive behavior in the control input. Again, 1000 simulations are done for each strategy in order to extract the mean of all values and also the variance.

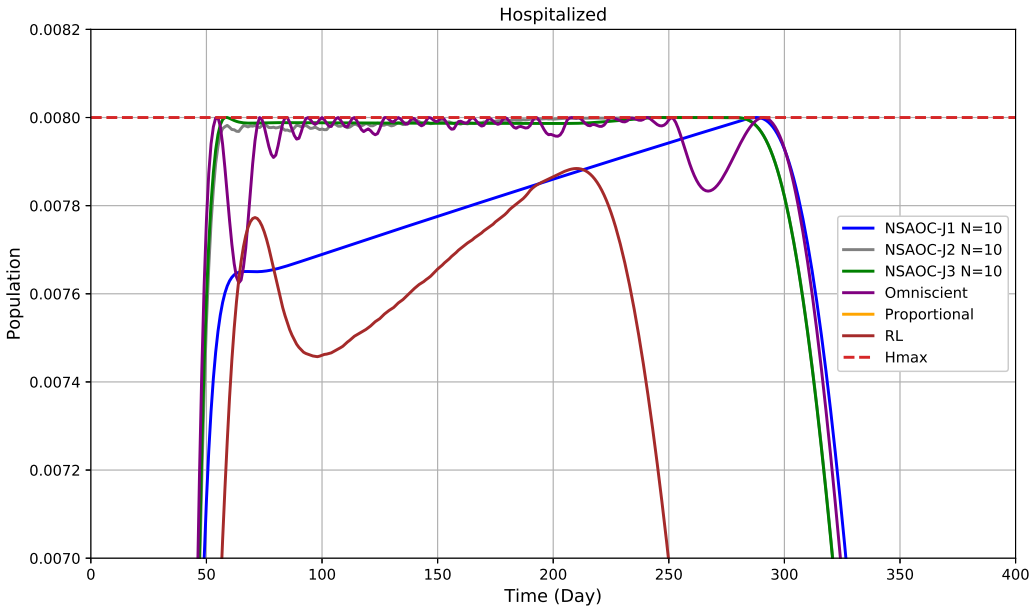

Figure 4.17: Hospitalization level graph amplified for different strategies using a SEIHRD model with constant parameters.

In Figure 4.18, the continuous lines represent the mean of each instant over all simulations, while the shaded areas represent the variance. The first conclusion is that the variance when using Omniscient Control is greater than all other strategies due to the fact of not having any feedback structure in its strategy.

The control effort graph shows that the mean values resultant from NSAOC strategies after instant $k = 60$ tend to decrease together around the same value. However, the variance is different for each objective function and, as we noted when using constant parameters, strategy NSAOC-J1 is smoother than NSAOC-J2 and NSAOC-J3. This is confirmed by the dashed blue area, which is smaller than the green and gray one.

The most important graph is the Hospitalization. Omniscient strategy is the worse one as expected, due to not having any feedback structure. PID-Like control keeps being the smoothest and safest strategy, securing the greatest slack to the limit. The last three strategies use NSAOC algorithm and the hospitalization levels are closer to H_{max} after they establish.

In Figure 4.19, we amplify the levels closer to H_{max} and also remove Omniscient and PID-Like strategies, since they do not lie closer to this level. First, even the mean values exceeds H_{max} using strategy NSAOC-J2, with the worst case almost reaching 0.009. Strategy NSAOC-J3 has their mean values below H_{max}, however in some cases the hospitalization level also exceeds its desired capacity limit, which can be noted by the dashed green area around instant $k = 45$. The same occurs

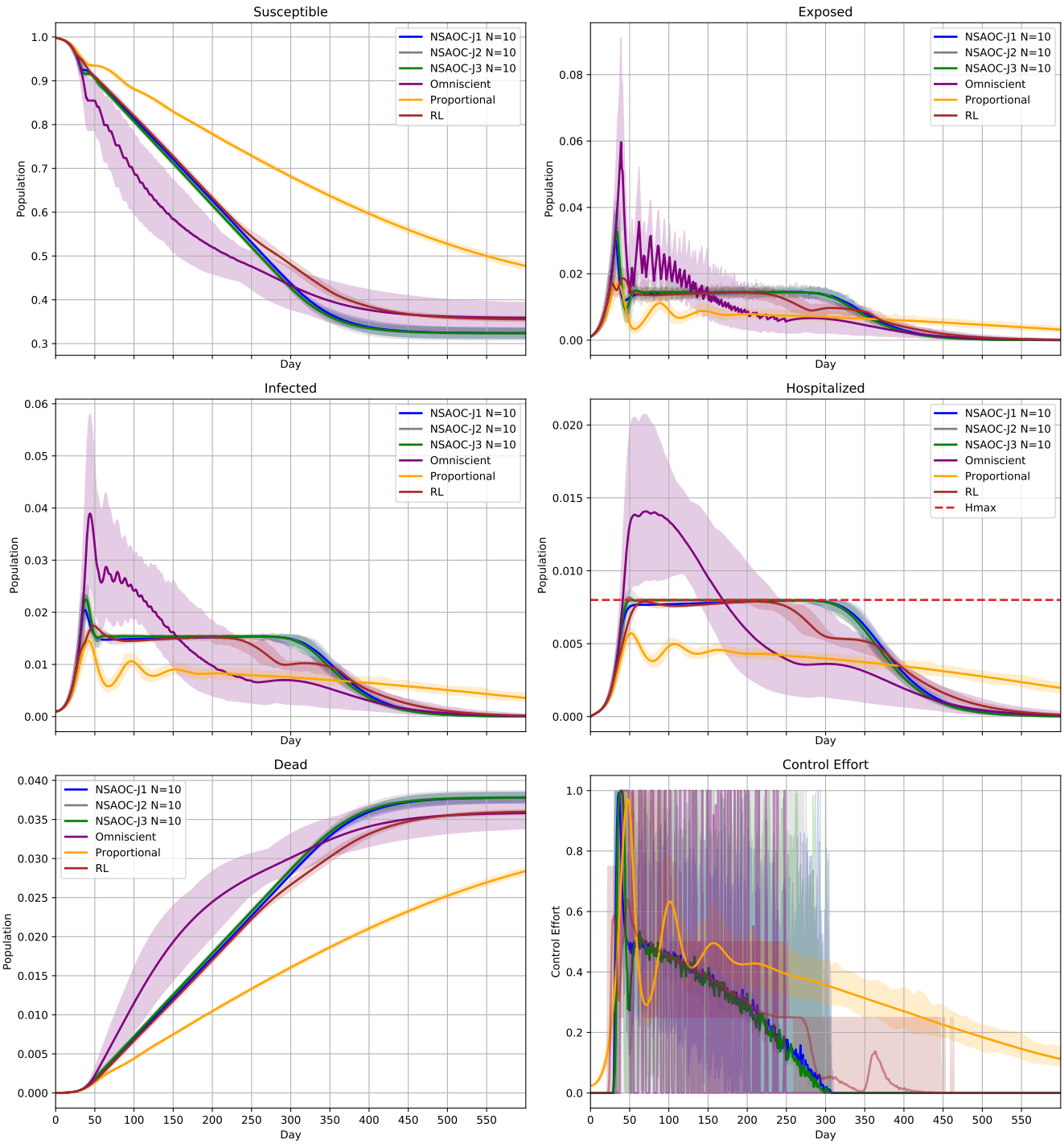

Figure 4.18: Results of a SEIHRD model with parameter uncertainty using all different strategies.

for strategy RL, that in most cases the curve is below the line, except for a small dashed area around instant $k = 100$. The variance is also high due to the inability to adapt to a model with different parameter values. Finally, strategy NSAOC-J1 is always below H_{max}, even in the worst case scenario. However, this fact has a small cost, being the strategy with the third higher total control input, as listed in Table 4.8.

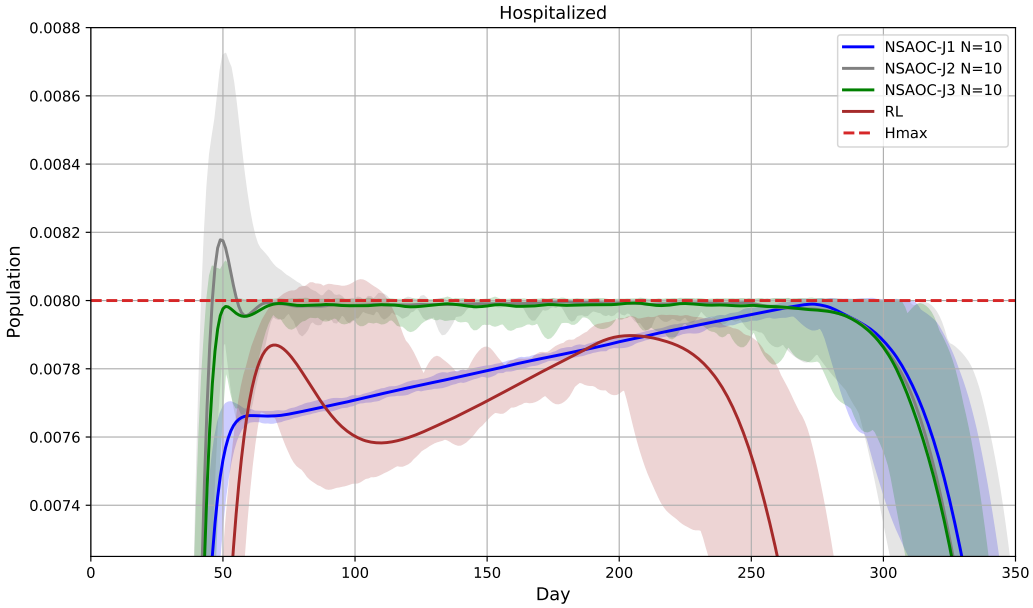

Figure 4.19: Hospitalization level graph amplified with strategies NSAOC-J1, NSAOC-J2, NSAOC-J3 and RL using a SEIHRD model with parameter uncertainty.

Strategy	Max(I)	Max(E)	Max(H)	Deaths	Sum(u)
NSAOC-J1	2.049%	3.002%	0.799%	3.781%	90.26
NSAOC-J2	2.326%	3.422%	0.818%	3.786%	85.15
NSAOC-J3	2.249%	3.257%	0.799%	3.782%	87.4
Omniscient	3.895%	5.963%	1.407%	3.589%	70.82
Prop.	1.457%	1.708%	0.571%	3.305%	213.24
RL	1.742%	1.862%	0.79%	3.609%	102.12

Table 4.8: Mean values of all simulations using all strategies in a SEIHRD model with parameter uncertainty.

Thus, considering that parameters vary in a real world, the best strategy seems to be NSAOC-J1, as it guarantees the hospitalization level below H_{max} with a total control effort slightly superior to other similar strategies.

Chapter 5

Vaccination

In the previous chapter, simulations only considered one control variable that lowers the contact between individuals carrying the virus (i.e. infected and exposed population) and susceptible individuals.

In this chapter, we introduce a new control variable. The vaccination variable affects directly the susceptible state, making the individuals from this group acquire immunity without having to be infected. They go directly to the Recovered group. The new control schema is shown in Figure 5.1.

After adding a new control variable, the control problem is changed. In fact, several control problems turn controllable after adding a new control variable to it.

The new proposed model is very similar to the one presented in Chapter 3 (Figure 3.1). The only difference is the additional vaccination variable in the equation of the susceptible individuals:

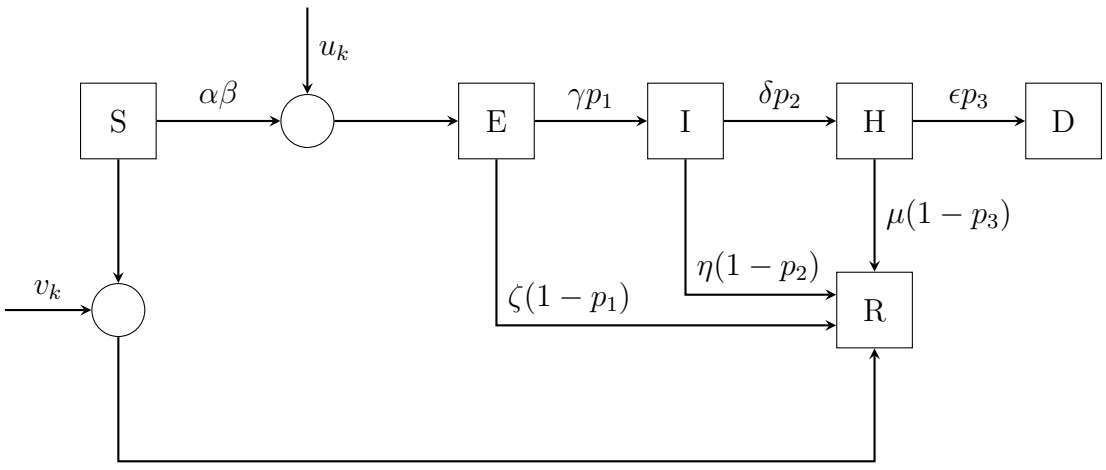

Figure 5.1: SEIHRD model diagram with 2 control variables: the NPIs level and vaccination.

$$S_{k+1} = S_k - (1-u_k)(\alpha S_k E_k + \beta S_k I_k) - v_{k-d_1} \tag{5.1}$$

$$E_{k+1} = E_k + (1-u_k)(\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1-p_1))E_k \tag{5.2}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1-p_2))I_k \tag{5.3}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1-p_3))H_k \tag{5.4}$$

$$R_{k+1} = R_k + \zeta(1-p_1)E_k + \eta(1-p_2)I_k + \mu(1-p_3)H_k + v_k \tag{5.5}$$

$$D_{k+1} = D_k + \epsilon p_3 H_k \tag{5.6}$$

In equation (5.1), v_{k-d_1} is the vaccination rate applied on the population at instant k and d_1 is the duration to the vaccine takes effect, considering that one shot gives full immunity.

In order to compare the effects of the vaccination, we use strategy NSAOC with performance index J_1 and parameter $N = 10$. This strategy was used in the previous chapter without any vaccination plan and it performed well, even with parameter uncertainty. So, the optimization problem we aim to solve in this chapter is:

$$\text{min} \qquad J = \sum_{k}^{k+N} u_k \tag{5.7}$$

$$\text{subject to:} \quad S_{k+1} = S_k - (1-u_k)(\alpha S_k E_k + \beta S_k I_k) - v_{k-d_1} \tag{5.8}$$

$$E_{k+1} = E_k + (1-u_k)(\alpha S_k E_k + \beta S_k I_k) - (\gamma p_1 + \zeta(1-p_1))E_k \tag{5.9}$$

$$I_{k+1} = I_k + \gamma p_1 E_k - (\delta p_2 + \eta(1-p_2))I_k \tag{5.10}$$

$$H_{k+1} = H_k + \delta p_2 I_k - (\epsilon p_3 + \mu(1-p_3))H_k \tag{5.11}$$

$$H_k \leq H_{max} \tag{5.12}$$

$$0 \leq u_k \leq 1 \tag{5.13}$$

Several vaccination strategies can be used. An optimal daily vaccination strategy is proposed in [13]. They established an optimal control problem to design vaccination strategies where vaccination modulates dynamics susceptibility through an imperfect vaccine. However, in this work we apply constant vaccination rate in order to analyze the effects of vaccination in the total control input and the states of the SEIHRD model.

From Figure 5.2, the total effort to control the epidemic decays when the vaccination rate is increased. When this rate is greater than 1.8%, there is no need to apply any kind of lockdown on the population. This would be equivalent of vaccinating 3.8 million individuals per day in Brazil, for example.

The evolution of the control effort and the number of susceptible individuals for

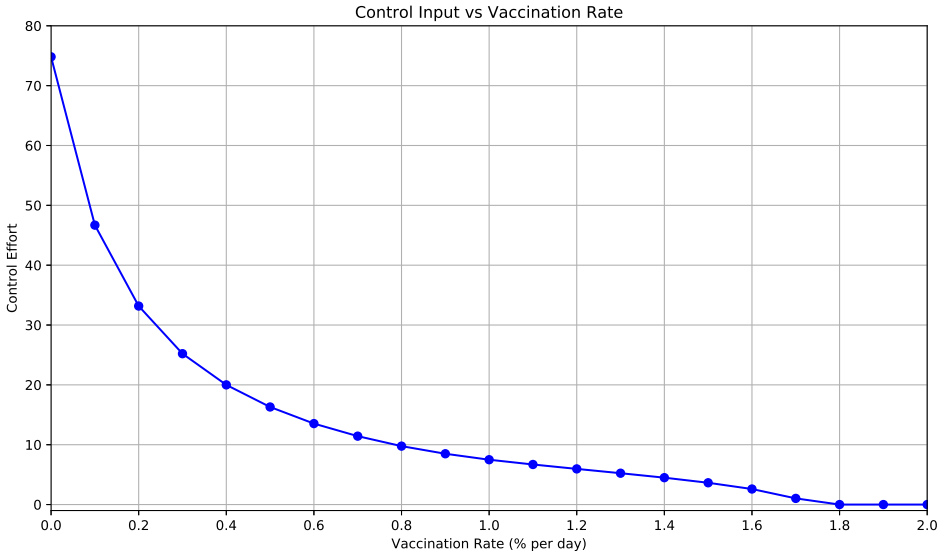

Figure 5.2: Total control input and variant vaccination rate using NSAOC ($N = 10$) in a SEIHRD model with constant parameters.

each vaccination rate is shown in Figure 5.3). In the first graph, we note that when the vaccination rate increases, it is possible to start acting ($u_k > 0$) with a higher delay, since there are fewer susceptible individuals at the same instant. Also, the peaks around the start of the period have lower values and they last for less time. In the second graph, the total number of susceptible individuals decays faster for higher vaccination rates, as expected. The last case ($v = 1.8\%$), the curve decay is constant, since the vaccination rate is constant and no control effort is applied in the population.

Similarly, the number of deaths decreases significantly with the first lower vaccination rates, as shown in Figure 5.4. As noted in table 5.1, even with a vaccination rate of 0.1%, it results in 1.2% less deaths compared to no vaccination plans. For higher vaccination rates, the differences are smaller, being less than 0.1% less deaths per additional 0.1% in the vaccination rate.

The hospitalization starts to decrease significantly when the vaccination rate is higher than 1.8%. This is related with the previous result that no NPIs are needed after this rate. So, as long as more individuals get vaccinated, less people will get infected and hospitalized, resulting in lower hospitalization levels. The numeric results for all vaccination rates are shown in table 5.1.

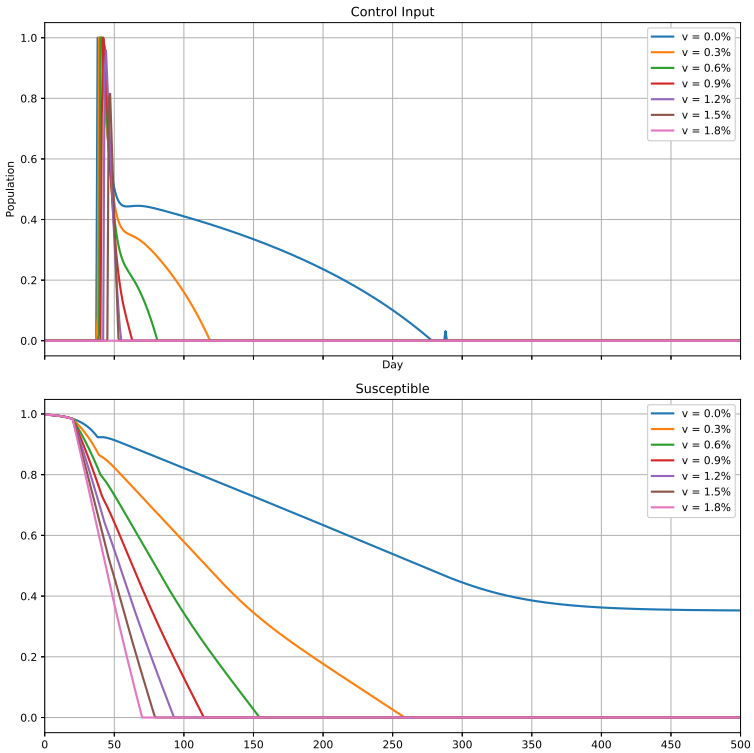

Figure 5.3: Control input and susceptible individuals using NSAOC-J1 ($N = 10$) in a SEIHRD model with constant parameters for each vaccination rate.

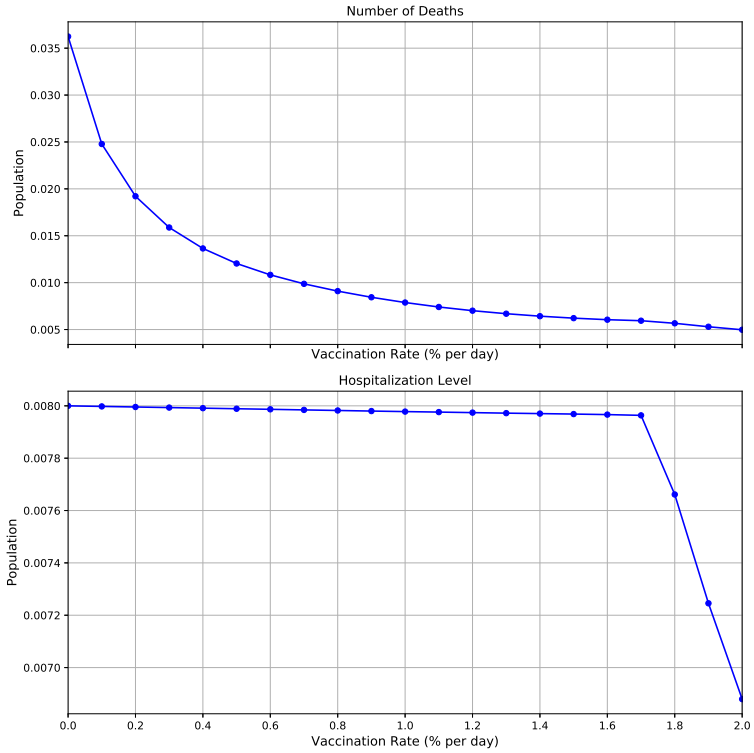

Figure 5.4: Number of deaths and hospitalization level using NSAOC-J1 ($N = 10$) in a SEIHRD model with constant parameters for each vaccination rate.

Vac. Rate	H	D	u
0.0	0.8%	3.624%	74.82
0.1	0.8%	2.478%	46.69
0.2	0.8%	1.921%	33.17
0.3	0.799%	1.588%	25.21
0.4	0.799%	1.365%	20.0
0.5	0.799%	1.204%	16.31
0.6	0.799%	1.083%	13.53
0.7	0.798%	0.987%	11.43
0.8	0.798%	0.91%	9.76
0.9	0.798%	0.844%	8.49
1.0	0.798%	0.788%	7.49
1.1	0.798%	0.74%	6.7
1.2	0.797%	0.701%	5.96
1.3	0.797%	0.669%	5.24
1.4	0.797%	0.643%	4.49
1.5	0.797%	0.621%	3.64
1.6	0.797%	0.605%	2.6
1.7	0.796%	0.594%	1.03
1.8	0.766%	0.567%	0.0
1.9	0.725%	0.53%	0.0
2.0	0.688%	0.498%	0.0

Table 5.1: Result values of a SEIHRD model with constant parameters for each vaccination rate.

Chapter 6

Conclusion

Different strategies are being used by governments all around the world involving the balance between public health and economic issues. In order to take such decisions, most of them rely on expert advice based on the results of epidemiological mathematical models and on daily case reports.

In this work, we explored different strategies that governments can use to control the COVID-19 epidemic. The main results from our analysis are the following:

- All strategies worked well when the SEIHRD model parameters are constant. They succeed on controlling the level of hospitalized people while minimized the total effort.
- Omniscient and NSAOC-J2 strategies resulted in a less total control effort, although they apply too many sequences of full lock down and no restrictions at all (like a On-Off policy). This can be difficult to apply in the real world.
- NSAOC-J1 and NSAOC-J3 presented a better behavior, since they imply a restricted lockdown at the beginning and start to relax as time goes on. However, the total control input is higher than using Omniscient Control.
- Reinforcement Learning and PID-Like Controller are safer strategies that take actions before the disease really spreads into the population, being easier to control the epidemic. However, they lead to higher total control inputs, with negative economic impacts.
- When the SEIHRD model parameters are not reliable enough, using any type of open loop control and, specifically, even the ideal Omniscient Control could result in not being able to control the level of hospitalized people, which can lead to a very high number of deaths.
- Only NSAOC-J1 and PID-Like strategies were able to keep the hospitalization level below the specified limit H_{max} in a SEIHRD model with parameter

uncertainty in all simulations. In most cases, NSAOC-J3 and RL strategies also succeed, as the mean values are below the limit.

- The number of deaths is proportional to the total control input aplied. Hence, reinforcement learning and PID-Like strategies had better results in this metric.

- In a SEIHRD model with parameter uncertainty, the number of deaths in all strategies increased due to the inability of predicting the future instants correctly.

- When the SEIHRD model parameters are not reliable, using Omniscient Control can result in not being able to control the level of hospitalized people, which can lead to a very high number of deaths. NSAOC strategies are able to control it due to their feedback structure.

- Adding vaccination results in less total control input, less deaths and smaller pandemic duration. For the first rate increases, the effects are more significant, leading to higher differences in the main performance indexes.

Many ideas can be further explored in future works:

- In our simulations, we considered daily strategies, which are difficult to apply in practice. Weekly strategies might be investigated corresponding to policies used in practice by several health autorities.

- The vaccination follows a constant rate in this work. However, it is possible to consider variable rates during time as presented in [13]. A different optimization problem could be studied to find an optimal vaccination strategy for SEIHRD model.

- Different performance indexes could be explored.

- A mapping relating total control input and real government actions could be created, indicating what represents each control level for the population.

Bibliography

[1] PAZOS, F., FELICIONI, F. E. "A control approach to guide nonpharmaceutical interventions in the treatment of Covid-19 disease using a SEIHRD dynamical model", *Complex Systems*, p. 24, 2020.

[2] KERMACK, W. O., MCKENDRICK, A. G. "A contribution to the mathematical theory of epidemics", *Proceedings of the Royal Society*, v. 115, pp. 700–721, 1927.

[3] GIORDANO, G., BLANCHINI, F., BRUNO, R., et al. "Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy", *Nature Medicine*, v. 26, pp. 1–6, 06 2020. doi: 10.1038/s41591-020-0883-7.

[4] "DAVID, J. D. E., BRAUER, F., VAN DEN DRIESSCHE, P., et al. *"Mathematical Epidemiology"*. Berlin Heidelberg, "Springer-Verlag", "2008".

[5] KEELING, M. J., ROHANI", P. *"Modeling Infectious Diseases in Humans and Animals"*. Princeton, N.J., "Princeton University Press", "2007".

[6] BACAËR", N. *"A Short History of Mathematical Population Dynamics"*. London, "Springer-Verlag", "2011".

[7] CARCIONE, J., SANTOS, J., BAGAINI, C., et al. "A Simulation of a COVID-19 Epidemic Based on a Deterministic SEIR Model", *Frontiers in Public Health*, v. 8, pp. 230, 05 2020. doi: 10.3389/fpubh.2020.00230.

[8] IVORRA, B., RUIZ FERRÁNDEZ, M., VELA, M., et al. "Mathematical modeling of the spread of the coronavirus disease 2019 (COVID-19) taking into account the undetected infections. The case of China", *Communications in Nonlinear Science and Numerical Simulation*, v. 88, pp. 105303, 04 2020. doi: 10.1016/j.cnsns.2020.105303.

[9] KOEHLER, J., SCHWENKEL, L., KOCH, A., et al. "Robust and optimal predictive control of the COVID-19 outbreak", *Annual Reviews in Control*, v. 51, 12 2020. doi: 10.1016/j.arcontrol.2020.11.002.

[10] BIN, M., CHEUNG, P., CRISOSTOMI, E., et al. "On Fast Multi-Shot COVID-19 Interventions for Post Lock-Down Mitigation", p. 19, 2020.

[11] WATKINS, N., NOWZARI, C., PAPPAS, G. "Robust Economic Model Predictive Control of Continuous-Time Epidemic Processes", *IEEE Transactions on Automatic Control*, v. PP, pp. 1–1, 05 2019. doi: 10.1109/TAC.2019.2919136.

[12] KAR, T., BATABYAL, A. "Stability analysis and optimal control of an SIR epidemic model with vaccination", *Bio Systems*, v. 104, pp. 127–35, 02 2011. doi: 10.1016/j.biosystems.2011.02.001.

[13] ACUÑA-ZEGARRA, M., DIAZ INFANTE, S., BACA CARRASCO, D., et al. "COVID-19 optimal vaccination policies: A modeling study on efficacy, natural and vaccine-induced immunity responses", *Mathematical Biosciences*, v. 337, pp. 108614, 05 2021. doi: 10.1016/j.mbs.2021.108614.

[14] ALMEIDA, L., BLIMAN, P.-A., NADIN, G., et al. "Final size and convergence rate for an epidemic in heterogeneous population", 10 2020.

[15] BLIMAN, P.-A., DUPREZ, M. "How Best Can Finite-Time Social Distancing Reduce Epidemic Final Size?" *Journal of Theoretical Biology*, v. 511, pp. 110557, 12 2020. doi: 10.1016/j.jtbi.2020.110557.

[16] BLIMAN, P.-A., DUPREZ, M., PRIVAT, Y., et al. "Optimal Immunity Control and Final Size Minimization by Social Distancing for the SIR Epidemic Model", *Journal of Optimization Theory and Applications*, v. 189, pp. 1–29, 05 2021. doi: 10.1007/s10957-021-01830-1.

[17] "CANON, M. *"Theory of Optimal Control and Mathematical Programming"*. "McGraw", "1970".

[18] "KIRK, D. *"Optimal Control Theory: An Introduction"*. "London", "Prentice-Hall", "1971".

[19] ISEE. "Stella Online - Covid Model". Disponível em: <`https://exchange.iseesystems.com/models/player/isee/covid-19-model`>.

[20] KE, R., ROMERO-SEVERSON, E., SANCHE, S., et al. "Estimating the reproductive number R0 of SARS-CoV-2 in the United States and eight European countries and implications for vaccination", *Journal of Theoretical Biology*, v. 517, pp. 110621, 2021.

[21] MENEZES MORATO, M., BASTOS, S., CAJUEIRO, D., et al. "An optimal predictive control strategy for COVID-19 (SARS-CoV-2) social distancing policies in Brazil", *Annual Reviews in Control*, v. 50, 07 2020. doi: 10.1016/j.arcontrol.2020.07.001.

[22] BELLMAN, R. *Dynamic Programming*. Princeton, Princeton University Press, 1957.

[23] BELLMAN, R. "A Markovian Decision Process", *Journal of Mathematics and Mechanics*, v. 6, n. 5, pp. 679–684, 1957. ISSN: 00959057, 19435274. Disponível em: <http://www.jstor.org/stable/24900506>.

[24] SUTTON, R. S., BARTO, A. G. *Reinforcement Learning: An Introduction*. Cambridge, Massachusetts, London, England, A Bradford Book, 2018.

[25] FAN, J., WANG, Z., XIE, Y., et al. "A Theoretical Analysis of Deep Q-Learning". In: Bayen, A. M., Jadbabaie, A., Pappas, G., et al. (Eds.), *Proceedings of the 2nd Conference on Learning for Dynamics and Control*, v. 120, *Proceedings of Machine Learning Research*, pp. 486–489. PMLR, 10–11 Jun 2020. Disponível em: <https://proceedings.mlr.press/v120/yang20a.html>.

[26] MNIH, V., KAVUKCUOGLU, K., SILVER, D., et al. "Human-level control through deep reinforcement learning", *Nature*, v. 518, pp. 529–33, 02 2015. doi: 10.1038/nature14236.

[27] KOMPELLA, V., CAPOBIANCO, R., JONG, S., et al. "Reinforcement Learning for Optimization of COVID-19 Mitigation policies", 10 2020.

[28] BUSSELL, E., DANGERFIELD, C., GILLIGAN, C., et al. "Applying optimal control theory to complex epidemiological models to inform real-world disease management", *Philosophical Transactions of the Royal Society B: Biological Sciences*, v. 374, pp. 20180284, 07 2019. doi: 10.1098/rstb.2018.0284.

[29] CASELLA, F. "Can the COVID-19 Epidemic Be Controlled on the Basis of Daily Test Reports?" *IEEE Control Systems Letters*, v. PP, pp. 1–1, 07 2020. doi: 10.1109/LCSYS.2020.300991.

[30] MENEZES MORATO, M., NORMEY-RICO, J., SENAME, O. "Model Predictive Control Design for Linear Parameter Varying Systems: A Survey", *Annual Reviews in Control*, v. 49, 05 2020. doi: 10.1016/j.arcontrol.2020.04.016.

[31] ALAMO, T., GUTIÉRREZ, D., MILLÁN, P. "Data-Driven Methods to Monitor, Model, Forecast and Control Covid-19 Pandemic: Leveraging Data Science, Epidemiology and Control Theory", 06 2020.

[32] TARRATACA, L., DIAS, C., HADDAD, D., et al. "Flattening the curves: on-off lock-down strategies for COVID-19 with an application to Brazil", *Journal of Mathematics in Industry*, v. 11, 01 2021. doi: 10.1186/s13362-020-00098-w.

[33] BASTOS, S., CAJUEIRO, D. "Modeling and forecasting the early evolution of the Covid-19 pandemic in Brazil", *Scientific Reports*, v. 10, 11 2020. doi: 10.1038/s41598-020-76257-1.

[34] VENTURIERI, V., GONÇALVES, M., FUCK, V. "Mitigation of COVID-19 using social distancing of the elderly in Brazil: The vertical quarantine effects in hospitalizations and deaths", 01 2021. doi: 10.1101/2021.01.12.21249495.

[35] LYRA, W., DO NASCIMENTO, J.-D., BELKHIRIA, J., et al. "COVID-19 pandemics modeling with SEIR(+CAQH), social distancing, and age stratification. The effect of vertical confinement and release in Brazil", 04 2020. doi: 10.1101/2020.04.09.20060053.

[36] GANEM, F., MENDES, F., OLIVEIRA, S., et al. "The impact of early social distancing at COVID-19 Outbreak in the largest Metropolitan Area of Brazil", 04 2020. doi: 10.1101/2020.04.06.20055103.

[37] AFONSO, S., AZEVEDO, J., PINHEIRO, M. "Epidemic analysis of COVID-19 in Brazil by a generalized SEIR model", 05 2020.

[38] SCARABAGGIO, P., CARLI, R., DOTOLI, M., et al. "Model predictive control to mitigate the COVID-19 outbreak in a multi-region scenario", 11 2020. doi: 10.36227/techrxiv.13153154.v1.

[39] RAINISCH, G., UNDURRAGA, E., CHOWELL, G. "A dynamic modeling tool for estimating healthcare demand from the COVID19 epidemic and evaluating population-wide interventions", *International Journal of Infectious Diseases*, v. 96, 05 2020. doi: 10.1016/j.ijid.2020.05.043.

[40] MENEZES MORATO, M., PATARO, I., AMERICANO DA COSTA, M., et al. "Optimal Control Concerns Regarding the COVID-19 (SARS-CoV-2) Pandemic in Bahia and Santa Catarina, Brazil", 11 2020. doi: 10.48011/asba.v2i1.1673.

[41] SADEGHI, M., GREENE, J., SONTAG, E. "Universal features of epidemic models under social distancing guidelines", *Annual reviews in control*, v. 51, 04 2021. doi: 10.1016/j.arcontrol.2021.04.004.

[42] FAN, J., WANG, Z., XIE, Y., et al. "A Theoretical Analysis of Deep Q-Learning". 2020.