


## Article

# A Synthesis of Pulse Influenza Vaccination Policies Using an Efficient Controlled Elitism Non-Dominated Sorting Genetic Algorithm (CENSGA)

Asma Khalil Alkhamis \* and Manar Hosny 

College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

\* Correspondence: akalkhamis@ksu.edu.sa

**Abstract:** Seasonal influenza (also known as flu) is responsible for considerable morbidity and mortality across the globe. The three recognized pathogens that cause epidemics during the winter season are influenza A, B and C. The influenza virus is particularly dangerous due to its mutability. Vaccines are an effective tool in preventing seasonal influenza, and their formulas are updated yearly according to the WHO recommendations. However, in order to facilitate decision-making in the planning of the intervention, policymakers need information on the projected costs and quantities related to introducing the influenza vaccine in order to help governments obtain an optimal allocation of the vaccine each year. In this paper, an approach based on a Controlled Elitism Non-Dominated Sorting Genetic Algorithm (CENSGA) model is introduced to optimize the allocation of the influenza vaccination. A bi-objective model is formulated to control the infection volume, and reduce the unit cost of the vaccination campaign. An SIR (Susceptible–Infected–Recovered) model is employed for representing a potential epidemic. The model constraints are based on the epidemiological model, time management and vaccine quantity. A two-phase optimization process is proposed: guardian control followed by contingent controls. The proposed approach is an evolutionary metaheuristic multi-objective optimization algorithm with a local search procedure based on a hash table. Moreover, in order to optimize the scheduling of a set of policies over a predetermined time to form a complete campaign, an extended CENSGA is introduced with a variable-length chromosome (VLC) along with mutation and crossover operations. To validate the applicability of the proposed CENSGA, it is compared with the classical Non-Dominated Sorting Genetic Algorithm (NSGA-II). The results indicate that optimal vaccination campaigns with compromise tradeoffs between the two conflicting objectives can be designed effectively using CENSGA, providing policymakers with a number of alternatives to accommodate the best strategies. The results are analyzed using graphical and statistical comparisons in terms of cardinality, convergence, distribution and spread quality metrics, illustrating that the proposed CENSGA is effective and useful for determining the optimal vaccination allocation campaigns.

**Keywords:** Controlled Elitism Non-Dominated Sorting Genetic Algorithm (CENSGA); NSGA-II; variable-length chromosome (VLC); metaheuristic; multi-objective optimization; Pulse vaccination; allocation; scheduling; planning



**Citation:** Alkhamis, A.K.; Hosny, M. A Synthesis of Pulse Influenza Vaccination Policies Using an Efficient Controlled Elitism Non-Dominated Sorting Genetic Algorithm (CENSGA). *Electronics* **2022**, *11*, 3711. <https://doi.org/10.3390/electronics11223711>

Academic Editors: Juan Antonio Ortega, Cecilio Angulo and Luis Gonzalez Abril

Received: 15 October 2022

Accepted: 10 November 2022

Published: 13 November 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Influenza, also known as flu, is a seasonal disease that is responsible for diverse health complications that can lead to hospitalization or even death [1]. The dynamic nature of the flu virus is reflected in its mutability from season to season, as well as in the fact that individuals who contract the disease may develop different symptoms. With the introduction of vaccinations for different diseases starting from the nineteenth century, this remarkable era of vaccination has had a promising influence on public health and population growth [2]. Since then, a succession of well-planned campaigns have

been documented, for instance, to reduce or eradicate hepatitis, yellow fever, polio and measles [3]. Although vaccinations against contagious diseases are currently considered as an intuitive strategy to control an outbreak or epidemic, policymakers are still facing a big challenge in allocating vaccine doses to the public, especially for mutant diseases or pandemics, when the aim is eradicating the disease or keeping it within a tolerable level with minimal cost [4].

Vaccination strategy planning and optimization is a multi-disciplinary field in which mathematical models, epidemiological models, and decision-making are combined to allocate vaccine doses to the population. Major efforts in tackling vaccination allocation from multiple perspectives are usually defined using mathematical models, which can be solved with exact or approximate solution methods. In general, metaheuristic algorithms provide a useful tool that can help the decision-maker to realize an optimal (or near-optimal) solution within a reasonable time under multiple parameters, especially when the problem size is large and/or there exist a substantial number of underlying constraints. Nonetheless, the use of metaheuristics to develop an optimal vaccination allocation policy still has potential to be explored.

This work proposes a vaccination allocation policy using Pulse vaccination, based on an SIR (Susceptible–Infected–Recovered) epidemiological model that simulates the behavior of an infectious disease. A Pulse vaccination policy requires allocating vaccine doses in certain time steps, as well as vaccinating a predetermined proportion of the susceptible individuals, so that they become part of the recovered population, according to references [5–7]. This approach permits two types of control policies: heterogenous and homogeneous vaccination. Heterogenous vaccination involves conducting different sizes of Pulse control policies at arbitrary time periods. On the other hand, homogeneous vaccination refers to giving the same sizes of Pulse control vaccines at similar time periods. Both controls have multiple objectives, since the target is to minimize both the infection volume and the vaccination campaign costs. In this research, both types of Pulse vaccination are presented, since we formulate our problem as a two-phased Pulse vaccination that aims to model the flu dose allocation problem mathematically.

In every epidemic cycle, there are two distinct dynamic regimes: a guardian phase and a contingent phase. The guardian phase aims to keep the infection under control. The contingent phase requires a set of variant-timing vaccination policies on different quantities of susceptible individuals in order to form a complete optimal campaign. The idea is to synthesize optimal policy phases in order to reach a final combined optimal campaign. This allows for an effective controlling campaign, as shown in Section 5. The common methodology of vaccination found in the literature is based on a single vaccination policy used across the epidemic's time horizon, which is an inapplicable policy in reality, and tends to be far from optimal [8–10]. Therefore, a more compromised paradigm has been recognized in some other works, which involves a compromise between the effectiveness of the campaign to control the epidemic and the cost of its implementation. This methodology involved multi-objective genetic optimization approaches. The multi-objective paradigm has been enhanced by features such as local searches for better convergence, a hash table to avoid the re-evaluating of repeated solutions, and some adaptations in genetic operators.

Genetic algorithms (GAs) are population-based algorithms in which a set of solutions is considered in each generation. GAs mimic the behavior of natural evolution, which optimizes biological features using the principles of selection, crossover (recombination) and mutation. The essential component of GAs is reproduction, wherein a set of candidate solutions engages in a selection process; solutions with good genes within the population are selected, making up a set of parents for the next generation. From this set of relatively good solutions, the next generation is formed by applying the processes of crossover, and on some occasions, mutation, to produce a set of offspring. Both parent and offspring sets are then combined to form a new population that engages again in the selection process. Solutions are ranked by their state of being dominated by, or being dominant over, other solutions based on their fitness (objective function of the solution). In the final generation,

the best solutions are those belonging to the first front, which are not dominated by any other solution (known as the Pareto front) [11]. This paper deals with the question of finding an estimate of the Pareto-optimal set for the vaccination allocation problem using a Controlled Elitist NSGA-II (the Non-Dominated Sorting Genetic Algorithm) [12], which is a fast and elitist multi-objective optimization engine that makes selections considering both the non-dominance ranking and the crowding distance.

The research's novelty in this multi-disciplinary field stems from its design of an enhanced multi-objective metaheuristic approach, namely, a Controlled Elitist Non-Dominated Sorting Genetic Algorithm (CENSGA), for optimizing the strategy of planning the flu vaccine allocation. For similar optimization problems, CENSGA has yielded promising results in complex instances with recurrent values of objective coefficients [13]. Thus, the proposed research's adoption of CENSGA is inspired by the encouraging outcomes of the approach as applied to similar optimization domains. CENSGA has been identified as a robust optimization tool for reaching optimal parameters in diverse allocation problems, including supplier selection and order allocation [14], job-shop scheduling [15], the geometric configuration of parallel manipulators [16], hub location-allocation [17], and economic/emission load dispatch [18]. To tackle the vaccination allocation problem, the calibration of our model will be performed based on the related previous works [6], in order to approximate most of the problem parameters.

The remainder of this paper is organized as follows: In Section 2, related works are mentioned, followed by the introduction of some preliminaries in Section 3. After that, the problem description is formulated in Section 4, whereas the optimization model proposed to solve the vaccination allocation problem is explained in Section 5. In Section 6, parameter tuning is briefly discussed, while results and efficiency performance measures are given and discussed in Sections 7 and 8, respectively. Finally, the concluding and perspective remarks are given in Section 9.

## 2. Literature Review

A well-known attempt to address the distribution of influenza vaccine shots optimally using metaheuristic techniques was carried out by Patel et al. [19]. The nature of an outbreak is generally non-linear, complex, and stochastic, and so the problem is difficult to solve optimally using mathematical models. Thus, the researchers simulated a stochastic pandemic, wherein genetic algorithms (GA) and Random Mutation Hill Climbing (RMHC) were used to minimize morbidity and mortality rates in the population with a given availability of vaccine doses. The results demonstrated that, under different coverage settings, the policy of vaccine distribution had a significant impact, compared to random mass vaccination.

Another related problem associated with infectious diseases is heterogeneity in transmission spread, vulnerability and contact frequencies in different populations [20]. Hu et al. [21] attempted to maximize the effects of distributing vaccine doses to susceptible individuals among different age groups. A differential evolution (DE) algorithm was developed to guide the distribution of the available vaccines optimally. The effects of vaccination duration and vaccine coverage were considered. The objective was to reduce the prevalence of infection incidences in the corresponding population. Performance was examined using a classical epidemiology model (in this case, SEIR (Susceptible–Exposed–Infected–Recovered)). The results indicate that the DE algorithm was promising in terms of offering a vaccine distribution solution. However, the researchers emphasized that the effectiveness of the vaccination depends on choosing the right vaccination time and duration. To address these issues, the role of planning in vaccination campaigns is a critical area that must reach a compromise in relation to the intended factors [22].

In the study of da Cruz et al. [6], the researchers considered an evolutionary multi-objective optimization algorithm and a stochastic simulation of the SIR epidemic dynamics. A bi-objective genetic algorithm, NSGA-II, was proposed as a way to lower control costs, as well as the number of infected individuals. The optimization process consisted of

two stages, in which a complete set of policies is formed by applying different styles of heterogenous vaccination followed by homogenous policies to reduce the integral infection at first, and then keep the infection at below harmful levels. The optimization algorithm was combined with a quadratic approximation as a local search, as well as a hash table for information storage. The results were proven to be statistically significant compared to the conical form of NSGA-II. This work is closely related to our work in this paper.

One major contribution of our study is introducing a modified GA with a variable-length chromosome (VLC) extension to address the problem of the time-varying polices employed in the contingent phase. The most demanding issues when modeling VLC are the adaption of the GA processes and evaluating the fitness function. Numerous variations in VLCs have been introduced in the literature for several problems. In [23], the authors suggested a VLC to represent QoS-aware multi-path web-service composition plans in order to find the optimal path using GA. In their work, the customer service requirements are represented as a directed graph, and the service selection and composition problem is translated into a pattern-matching problem of service travel flow. The crossover process is updated and programmed by cut and splice models on the graph paths, while the mutation process remains as it is, similar to other GA processes. VLC has also been used on planning problems. For example, the authors in [24] proposed a VLC for a system that considers planning dilemmas, i.e., finding a good sequence of actions that satisfies the provided objectives. In addition, in the graph-based paradigm, Cruz-Piris et al. [25] modeled a VLC to address a road traffic coordination multi-path problem, in which the genes of the chromosome encode the path branch. VLC was also operated on the complex chromosome in genetic programming (GP) [26], which is a peer evolutionary technique used to find an optimal computer program. However, this approach was designed for problems represented in a tree-based structure only. In [27], the authors proposed a unique pixel-level Multi-Spectral (MS) very high-resolution (VHR) image segmentation algorithm based on variable-length NSGA-II clustering. In their approach, they maintained centroids for segmentation via gene value, which implies a flexible chromosome style used to address the shrinking or extending of chromosome length, without comprising the quality of the segmentation results. Adding this property with the necessary updating of crossover and mutation processes helped to amend the optimization's main drawbacks; a similar approach is employed in our work.

### 3. Preliminaries

#### 3.1. Multi-Objective Evolutionary Optimization

The multi-objective evolutionary algorithm (MOEA) is a stochastic optimization technique used to solve many NP-hard real-world optimization problems [28,29]. The mathematical definition of a multi-objective problem (MOP) can be formulated as (assuming a minimization problem):

$$\text{minimize } F(x) = (f_1(x), f_2(x), f_3(x), \dots, f_m(x))^T \text{ subject to } x \in \Phi \quad (1)$$

The function  $F: \Phi \rightarrow R^m$  maps  $x \in \Phi$  decision vector to  $M$  real-valued objective functions. Here,  $\Phi$  is the decision (variable) space and  $R^m$  represents the objective space [30]. The objectives in Equation (1) predominately collide with each other. Finding a single solution that optimizes all objectives at once is usually not feasible, since the improvement of one objective may cause the deterioration of another [28]. Therefore, an MOEA aims to find a solution set by employing the concept of domination to find optimal Pareto solutions [29]. Pareto-optimal solutions are the best trade-off solutions in the population, obtained by calculating the objective value for each decision vector, which is used to determine the rank of dominance in the population; it then selects a potentially better solution to join the reproduced population [28,29]. The concepts of dominance and Pareto-optimality are formally defined as follows [28,30]:

**Definition 1.** A vector  $X_1 = (x_{1,1}, \dots, x_{1,n})^T$  is said to dominate another vector  $X_2 = (x_{2,1}, \dots, x_{2,n})^T$ , denoted as  $X_1 \prec X_2$  iff  $\forall i \in \{1, \dots, n\}, x_{1,i} \leq x_{2,i}$  and  $X_1 \neq X_2$ .

**Definition 2.** A feasible solution  $x_1 \in \Phi$  of the problem in Equation (1) consisting of all non-dominated solutions is called a Pareto-optimal solution, iff  $\nexists x_2 \in \Phi$  such that  $F(x_2) \prec F(x_1)$ .

**Definition 3.** A Pareto set (PS) consists of the collection of all the Pareto-optimal solutions, denoted as

$$PS = \{x_1 \in \Phi \mid \nexists x_2 \in \Phi, F(x_2) \prec F(x_1)\}.$$

The image of the PS in the objective space is called the Pareto front (PF)

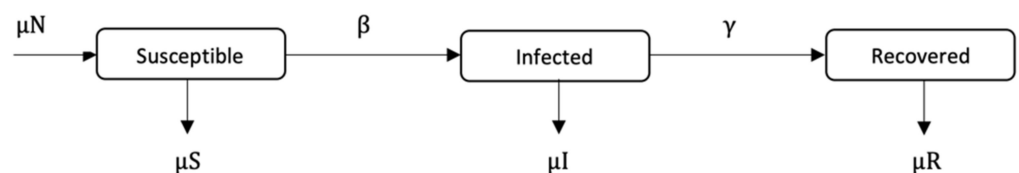
$$PF = \{F(x_1) \mid x_1 \in PS\}.$$

### 3.2. Epidemiological Models

Epidemiological models examine the spread of infectious diseases in a population over a time period, by classifying the individuals in a set of disjoint classes. An individual within a certain time unit is recognized on one class only. Studying individuals' behavior and their movements between compartments helps in assessing and forming appropriate procedures and policies to override or mitigate the effects of diseases as much as possible by vaccination, for example, or by applying any countermeasure means [7,31]. The following section describes the compartmental model used in this work, the SIR (Susceptible–Infected–Recovered) model.

#### SIR Model

The epidemiological model SIR was first outlined by Kermack and McKendrick in 1927 [32]. Their contribution is still the most widely used to illustrate infectious disease dynamics. The SIR model depicts the spread behavior of an infectious disease. For a contagious virus such as influenza, the transmission dynamic comes in stages. It starts when individuals susceptible to contamination with the virus particle become infected after the incubation time passes. Subsequently, when the infectious individual has recovered and gained immunity, they move into the recovered class. If there is a reliable vaccine for the virus, the susceptible individual is moved instantly to the recovered class, without passing through the infected class, after the vaccine immunizes that individual [6,8] (Figure 1 demonstrates the SIR normal dynamics).



**Figure 1.** General outline of SIR model.

Mathematically, the SIR model uses three differential equations to describe an infectious disease. The initial value problem is presented in Equation (2). Here,  $S$ ,  $I$  and  $R$  represent the numbers of individuals in each compartment, respectively: susceptible, infected and recovered.  $N$  is the total number of individuals in the population (it must be constant on each time unit  $t$ :  $S(t) + I(t) + R(t) = N, \forall t \geq 0$ ). The parameters' descriptions

and their values are listed in Table 1. As  $N$  is supposed to be fixed across time, the birth rate and mortality rate are made equal [6].

$$\begin{aligned} \frac{dS}{dt} &= \mu N - \mu S - \frac{\beta IS}{N}, S(0) = S_0 \geq 0, \\ \frac{dI}{dt} &= \frac{\beta IS}{N} - \gamma I - \mu I, I(0) = I_0 \geq 0, \\ \frac{dR}{dt} &= \gamma I - \mu R, R(0) = R_0 \geq 0, \end{aligned} \tag{2}$$

**Table 1.** Model parameters.

Parameter	Definition	Value
$\beta$	Transmission rate	$2.36 \text{ (t.u.)}^{-1}$
$\gamma$	Recovery rate of infected individuals	$1/7 \text{ (t.u.)}^{-1}$
$\mu^1$	Birth and mortality rate	$1/7 \text{ (t.u.)}^{-1}$
$R_0$	Reproduction number	15
$i_{tol}$	Tolerance ratio	0.01

<sup>1</sup> All newborns have no birth immunity against flu. Additionally, according to the CDC estimation between 2010 and 2020, flu is annually responsible for 9 million–41 million illnesses, 140,000–710,000 hospitalizations and 12,000–52,000 deaths. <https://www.cdc.gov/flu/about/burden/index.html> [accessed on 1 November 2022].

A key number integrated with the epidemiological model is the reproduction number ( $R_0$ ); it represents the number of secondary cases generated from exposure to a primary individual [33]. The  $R_0$  value is determined by the epidemiological model used; for instance, for SIR,  $R_0 = \beta / (\mu + \gamma)$  [34]. The value of  $R_0$  is fixed and reflects the necessity of employing vaccination or any countermeasure strategies. If  $R_0 < 1$ , then the disease dies out, and henceforth, tends towards an “infection-free equilibrium” without intervention; here, the endemic cannot hold the growth since each infected individual on average infects less than one member of the population. However, if  $R_0 > 1$ , then the disease invades the population and stabilizes at an “endemic equilibrium”. In fact, the disease then turns to an endemic state, where the disease self-sustains in the population at a certain level, having the ability to propagate continuously [34,35]. The only way to diminish the effect of an infectious disease with  $R_0 > 1$  is by introducing a relevant countermeasure strategy, which in this study is vaccination. Another number that is key to the vaccination process is known as the effective reproduction number ( $R_{EF}$ ). This represents the number of secondary cases generated from exposure to a primary individual during vaccination. The value of  $R_{EF}$  can be updated with the progress of vaccination, unlike  $R_0$ . Therefore, reaching the state of  $R_{EF} < 1$  is desirable in marking the end of the contiguous disease and halting the vaccination process. The value of  $R_{EF}$  can be derived from  $R_0$ , by  $R_{EF} = s(t) \cdot R_0$  [36].

Using a specific time unit (t.u.), the period defined to determine a complete vaccination campaign is  $T_{ctrl} = T_{tmp} + T_{gc} = 150$  (t.u.). At time unit 50, the system reaches “endemic equilibrium”, which is marked as  $T_{tmp} = 50$  (t.u.), and this separates the time periods into before and after the infection stabilized. From the beginning of endemic equilibrium at  $T_{tmp}$  to  $T_{ctrl}$ , the optimization system is dedicated to finding a guardian control policy, while during the time period from  $\tau_0$  to  $T_{tmp}$ , the optimization system operates under the contingent control phase to find a set of policies that are able to contain the infection, as shown in Figure 2. The guardian phase is better applied indefinitely, but since this is an unrealistic situation, it will be considered over a time period of length  $T_{gc}$ . The initial SIR condition for the contingent control is supposed to be  $(s_0, i_0, r_0) = (0.99, 0.01, 0)$ . However, for the guardian control, the initial system condition will be  $(s_e, i_e, r_e) \approx (0.067, 0.085, 0.848)$ . Remarkably, the value of  $i_e = 0.085$  is greater than the acceptable tolerance ratio  $i_{tol} = 0.01$ , so this is a clear justification of the importance of deploying the guardian control policy in order to minimize the infection volume after completing the contingent control policies. The setup of SIR parameter values in this context mimics a hypothetical disease that is hard to control but for which there is an efficacious vaccine (refer to Table 1).

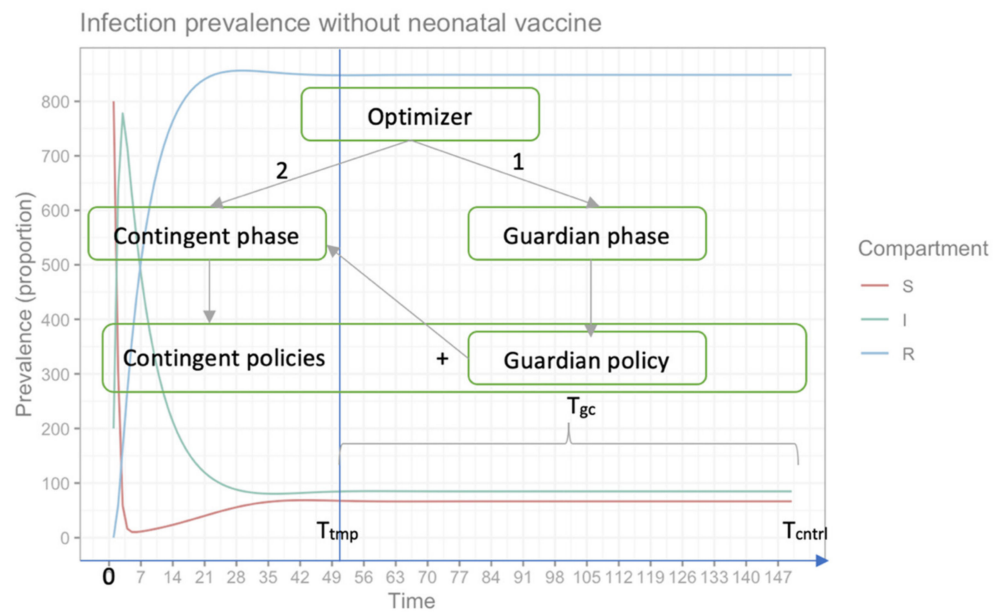


Figure 2. Anatomy of the optimizer’s main components.

#### 4. Problem Description

The role of vaccination allocation is a paramount aspect in controlling the spread of infectious diseases, especially influenza and other contagious viruses. The effect of vaccination in reducing the fraction of susceptible individuals in order to control the outbreak/epidemic at an early point in time is directly related to avoiding or reducing an increase in the overall infection volume. A two-fold effect of vaccination is well recognized: a direct advantage for the vaccinated, due to acquiring full or partial immunity, and an indirect benefit for the unvaccinated by reducing the risk of exposure to the disease [37].

##### Pulse Vaccine Allocation

This work is based on Pulse vaccination, which means that in a predetermined time period, there is a vaccination policy that will be applied on a certain percentage of susceptible individuals. This approach allows different/exact sizes of Pulse control actions at arbitrary/autonomous time points in order to form a complete vaccination campaign [3,6,7]. According to the work of A. C. S. DUSSE et al. [3], applying a deterministic (discrete) or stochastic (continuous) SIR system in Pulse vaccination yields negligible differences in the obtained findings. Moreover, the effects of Pulse vaccination on a continuous system can be observed as a transformation of the continuous system to a non-continuous system. The Pulse action affects the functions (S), (I), and (R) by turning them into step functions [3]. Thus, the aim of this research is to study Pulse vaccination control from a discrete time optimization point of view.

Let a vaccination time horizon defined by  $T \in \mathbb{R}$  and  $\Omega = \{\tau_0, \dots, \tau_N\}$  represent a set of pivot points in the closed time interval  $[0, T]$ , such that:  $\tau_k < \tau_{k+1}$ ,  $\tau_0 = 0$  and  $\tau_N = T$ . These timeframes do not have to be equidistant. Consider an epidemiological model compartment  $x$  be a state variable belonging to  $\Phi$ , and the control vaccination rates belonging to  $U_k = \{v_0, \dots, v_k\}$ , for all  $k = 0, \dots, N - 1$ . The system state at instance  $\tau_k$  is denoted by  $x(\tau_k)$  and the control action is denoted by  $u[k]$ . The instant of time that comes immediately after  $\tau_k$  is  $\tau_{k+1}$ , and is defined as a time instant “just after” the Pulse action in  $\tau_k$  [5]. In short, the Pulse control problem is defined formally by T. Yang in [38] as follows:

**Definition 4.** In a Pulse control problem, the state at each time  $\tau_k$ ,  $x(\tau_k) \in \Phi$  can be changed by  $x(\tau_k^+) = x(\tau_k) + u[k]$ , with  $u[k] \in U_k$ .

The dynamic of the system under the effect of Pulse control can be outlined in this form:

$$\begin{aligned} x(t+1) &= f(t, x(t)), \\ x(\tau_k^+) &= x(\tau_k) + u[k], \\ t &\in (\tau_k^+, \tau_{k+1}], \\ k &= 0, \dots, N-1, \end{aligned} \quad (3)$$

Activating the control action (vaccination policy) at each timeframe establishes a new initial condition for the system to be the input of the subsequent timeframe. A complete vaccination campaign is constituted by linking each timeframe outcome with the next according to Equation (3); the whole process is in line with “Bellman’s Optimality Principle”: an optimal solution (set of policies) has the property, no matter of its initial policy, that the remaining policies must form an optimal solution from that initial policy [39].

## 5. Optimization Model

### 5.1. Proposed Model

This paper pursues multi-objective pulsive vaccination control using an open-loop continuous variable dynamic optimization procedure to solve the epidemiological problem, using the SIR model as the dynamic system and two multi-objective optimization techniques, NSGA-II and CENSGA, to be described in Section 5.2. Optimization is one of the most active and effective techniques used to control vaccination policies. The limited number of vaccine doses added to time constraints overwhelms the infection volume. A cost-effective vaccination campaign is crucial to mediate between the reductions in infection volume and the cost of vaccination administration.

The mathematical formulation of the problem is described as follows. Let  $x \in \Phi$  constitute the decision variable vector;  $\Phi \subset R^n$  is the decision variable space;  $[f_1(x), \dots, f_m(x)]$  is the vector of objective functions to be minimized and  $X^*$  is the set of optimal solutions, which consist of all decision vectors for which the corresponding objective vector cannot be ameliorated in any dimension without a recession in another one (recall the definitions in Section 2). In the optimization set-up, we must determine the chromosome representation and fitness equation that helps in examining the objective functions. For the sake of this work, we adopt the general framework methodology found in [3,6,7] to test the suggested CENSGA technique under the same conditions and parameter calibrations.

#### A. Fitness Function

The fitness function contributes to determining the quality of a given chromosome (solution), and it returns numerical value(s) reflecting the evaluation status of the instance. In determining the fitness equation, it is necessary to combine the objectives and constraints to obtain a set of feasible chromosome instances. Essentially, the goal here is minimizing the volume of an infected population over time and, at the same time, to obtain a campaign that offers an effective cost within the vaccination policies [3,6].

The optimization process has two phases: the guardian control determines a set of non-dominated guardian policies, and the contingent control determines a set of contingent policies. The outcome of the contingent phase is complemented with the best guardian control policy selected from the guardian phase to form a complete vaccination campaign. The purpose of two-phased optimization is to maintain an acceptable level of infected individuals in a finite time near to the disease-eradication ratio (i.e., compartment  $(I) \leq \textit{tolerance ratio} (i_{tol}) > 0$ ), which could be achieved by incorporating a fixed-guardian control policy following a contingent vaccination phase in order to avoid an/another outbreak.

The cost function employed as the second objective function follows the exact cost function defined by Cruz et al. [6]. For the sake of comparison, the same simulation parameters and calibration values in [6] are used here as well. There are fixed and variable cost parameters that contribute to the implementation of vaccination policies. Starting with the fixed-cost part, the constants shown in Equation (4) are assumed to be  $c_1 = 10$ ,  $c_2 = 1$



and  $c_3 = 1$ . Thereby,  $c_1 \cdot N_{gc}$  is the fixed part of the cost function with the implementation of all policies in the guardian control phase, i.e.,  $N_{gc}$  is the number of times the selected guardian control policy will be applied. The second term  $c_2 \cdot N_{gc} \cdot (1 + v_{gc})^2$  is the essence of the cost function that articulates the variable monetary cost related to the effort required to reach the  $v_{gc}$  proportion of the susceptible population in all policies. The last term,  $c_3 \cdot \sum v_{gc} \cdot S[k]$ , stands for the cost of all vaccines in all vaccination policies. Without losing generality, the same definition principle is applicable for the cost function of the contingent control phase found in Equation (6), where  $N_{cc}$  indicates the number of contingent control policies that are planned to be administrated.

Mathematically, the time considered to determine the multi-objective control period is  $T_{ctrl} = T_{imp} + T_{gc}$ , where  $T_{ctrl}$  belongs to the closed interval  $[0, T_{ctrl}]$ , and has to be partitioned into a set of events  $\Omega = \{\tau_0, \dots, \tau_N\}$ , such that:  $\tau_0 = 0$ ,  $\tau_N = T_{ctrl}$ , and  $\tau_{k+1} - \tau_k = \Delta T$ . The time between each  $\tau_k$  and  $\tau_{k+1}$  is considered as a state of the optimization problem. The guardian phase of the multi-objective optimization model is depicted in Equation (4). The role of this phase is to find a set of non-dominated guardian control policies,  $X^*_{gc} = \{(\Delta\tau_{gc}, v_{gc}), \dots\}$ ; these are the decision variables used to derive a candidate solution with a constant time interval between campaigns,  $\Delta\tau_{gc}$ , and with a constant proportion of susceptible population, which must be vaccinated at each campaign,  $v_{gc}$ . The guardian control policy should begin at time  $T_{imp}$  and be applied indefinitely, up to disease eradication or, in our multi-objective period, until  $T_{ctrl}$ .

The multi-objective optimization model of the contingent phase is depicted in Equation (6). The role of this phase is to determine a non-dominated set of complete vaccination campaigns. This phase appoints a fixed-guardian control policy after a set of variable contingent control policies (refer to Figures 2 and 3). The set of non-dominated complete control policies for vaccination is described by

$$X^* = \{(\Delta\tau_1, v_1, \Delta\tau_2, v_2, \dots, \Delta\tau_{N_{cc}}, v_{N_{cc}}, \Delta\tau_{gc}, v_{gc}), \dots\}$$

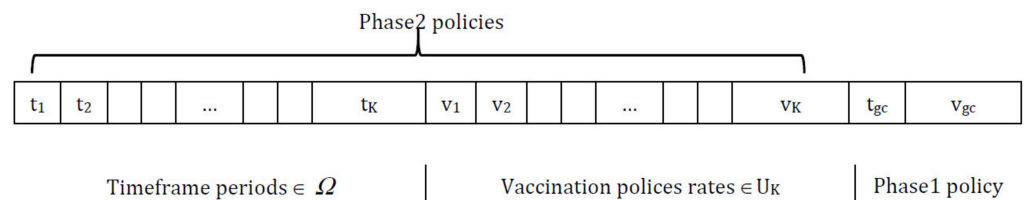


Figure 3. Chromosome representation.

Each contingent control is described by  $N_{cc}$  value pairs of the time interval between campaigns,  $\Delta\tau_k$ , and the fraction of susceptible individuals to be vaccinated,  $v_k$ ,  $k \in \{1, 2, \dots, N\}$ . The extra pair  $(\Delta\tau_{gc}, v_{gc})$  is derived from phase 1, which represents a guardian control policy chosen from solutions  $X^*_{gc}$ . The resulting optimization problem for the guardian control phase is described by Equation (6).

$$\min X_{gc} \begin{cases} F_1 = \int_{T_{imp}}^{T_{ctrl}} I(t) dt \\ F_2 = C_1 \cdot N_{gc} + C_2 \cdot N_{gc} (1 + v_{gc})^2 + C_3 \cdot \sum_{k=1}^{N_{gc}} v_{gc} \cdot S[k] \end{cases} \quad (4)$$

subject to (5):

$$\left\{ \begin{array}{l} \frac{dS}{dt} = \mu N - \mu S - \frac{\beta IS}{N}, \quad S(0) = S_0 \cdot N \geq 0 \\ \frac{dI}{dt} = \frac{\beta IS}{N} - \mu I - \gamma I, \quad I(0) = I_0 \cdot N \geq 0 \\ t_k = T_{tmp} + k \cdot \Delta\tau_{gc}, \tau \in (\tau_k^+, \tau_{k+1}] \\ s(\tau_k^+) = s[k^+] = s[k](1 - v_{gc}) \\ i(\tau_k^+) = i[k^+] = i[k] \\ i(t) \leq i_{tol} \cdot N, \quad t \in (T_{tmp}, T_{tmp} + T_{gc}] \\ \sum_{k=1}^{N_{gc}} \Delta\tau_k \leq T_{gc}; \quad N_{gc} = \lfloor \frac{T_{gc}}{\Delta\tau_{gc}} \rfloor \\ k = 0, 1, \dots, N_{gc} - 1 \\ 0.40 \leq v_{min} \leq v_{gc} \leq v_{max} \leq 0.95 \\ 1 \leq \Delta\tau_{min} < \Delta\tau_k < \Delta\tau_{max} \leq 20 \end{array} \right. \quad (5)$$

The guardian control has the following constraints, as described by Equation (5). The system dynamic is determined by the SIR model (see Equation (2)), starting at the endemic equilibrium point  $\{s(0), i(0), r(0)\} = \{0.067, 0.085, 0.848\}$ , which is the infection-free value of the SIR model under constant Pulse vaccination [6,35]; functions  $dS/dt$  and  $dI/dt$  behave accordingly. From the beginning of guardian control, vaccination moments occur periodically  $k_{th}$  times in a harmonic manner  $t_k = T_{tmp} + k \cdot \Delta\tau_{gc}$ . At each unit of time ( $t_k$ ), a constant Pulse vaccination policy  $v_{gc}$  (proportion of the susceptible to be vaccinated) is carried on, propagating the direct positive effect of vaccination on the population, as described in  $S(t_k^+)$ , and  $I(t_k^+)$ . The notation (+) indicates the moment after the vaccination carried out via the Pulse vaccination policy. To keep the infected population under control, during guardian control, the disease should remain below an acceptable value, by  $i(t) \leq i_{tol} \cdot N$ , where  $i(t)$  is an instance of  $dI/dt$  at time ( $t_k$ ). The number of times the guardian control policy is going to applied is determined by  $N_{gc}$ , taking into consideration the total sum of time periods, must not exceed  $T_{gc}$ . The proportion of susceptible population to be vaccinated in each policy is bounded in the range  $[v_{min}, v_{max}] = [0.40, 0.95]$ , while the time between two successful vaccination policies must be in the range  $[t_{min}, t_{max}] = [1, 20]$ . This search range includes values that theoretically prove that the infected population tends to zero over time [35]. The effects on compartment (R) are omitted for the sake of brevity, but this will not have any effects on the resulting output, as the system is an open system.

In contrast to the fixed-time policy applied during the guardian phase, time-variant control policies are considered for the contingent control phase, as described by Equation (6). The objective functions of the contingent phase are close to the ones anticipated in the guardian phase. In addition, the contingent phase objective functions consider the entire time horizon, which combines the contingent policies to stabilize disease fluctuation with the guardian policy to keep the infection volume under control. To sum up, a selected guardian control policy will be applied repeatedly on the time interval  $[T_{tmp}, T_{ctrl}]$ , while the contingent control policies are administrated on the time window  $[0, T_{tmp}]$ . In the contingent control phase, both policies' parts are connected to form a complete vaccination campaign that operates on the entire optimization time horizon  $[0, T_{ctrl}]$ .

$$\min X_{cc} \left\{ \begin{array}{l} F_1 = \int_0^{T_{ctrl}} I(t) dt \\ F_2 = C_1 \cdot N_{Total} + C_2 \cdot \left( \sum_{k=1}^{N_{gc}} (1 + v_k)^2 + N_{gc} (1 + v_{gc})^2 \right) + C_3 \cdot \sum_{k=1}^{N_{Total}} v_k \cdot S[k] \end{array} \right. \quad (6)$$

subject to (7):

$$\left\{ \begin{array}{l}
 \frac{dS}{dt} = \mu N - \mu S - \frac{\beta IS}{N}, \quad S(0) = S_0 \cdot N \geq 0 \\
 \frac{dI}{dt} = \frac{\beta IS}{N} - \mu I - \gamma I, \quad I(0) = I_0 \cdot N \geq 0, \quad \tau \in (\tau_k^+, \tau_{k+1}] \\
 t_k = \tau k - 1 + \Delta \tau k \\
 s(\tau_k^+) = s[k^+] = s[k](1 - v_k) \\
 i(\tau_k^+) = i[k^+] = i[k] \\
 v_k = \begin{cases} v_k, & 1 \leq k \leq N_{cc} \\ v_{gc}, & N_{cc} + 1 \leq k \leq N_{gc} \end{cases} \\
 \Delta \tau k = \begin{cases} \Delta \tau k, & 1 \leq k \leq N_{cc} \\ \Delta \tau_{gc}, & N_{cc} + 1 \leq k \leq N_{gc} \end{cases} \\
 i(t) \leq i_{tol} \cdot N, \quad t \in (T_{tmp}, T_{tmp} + T_{gc}] \\
 \sum_{k=1}^{N_{gc}} \Delta \tau k \leq T_{gc}; \quad N_{gc} = \lfloor \frac{T_{gc}}{\Delta \tau_{gc}} \rfloor \\
 \sum_{k=1}^{N_{cc}} \Delta \tau k \leq T_{tmp}; \quad N_{Total} = N_{gc} + N_{cc}; \\
 N_{cc} \in \left\{ \left\lfloor \frac{T_{tmp}}{\Delta \tau_1} \right\rfloor, \dots, \left\lfloor \frac{T_{tmp}}{\Delta \tau_k} \right\rfloor \right\}, \quad \sum \Delta \tau k \leq T_{tmp} \\
 k = 0, 1, \dots, N_{gc} - 1 \\
 0.40 \leq v_{min} \leq v_k \leq v_{max} \leq 0.95 \\
 1 \leq \Delta \tau_{min} < \Delta \tau k < \Delta \tau_{max} \leq 20
 \end{array} \right. \tag{7}$$

The contingent phase combines guardian and contingent policies to form a final vaccination campaign. This considers the optimization problem in the contingent phase as an extension of the former optimization problem with their mutual set of constraints, as described by Equation (7). The system still behaves dynamically using the SIR model with the same parameters as before. The initial state of susceptible, infected and recovered populations is assumed to be  $\{s(0), i(0), r(0)\} = \{0.80, 0.20, 0\}$ . Each  $k$  policy is indicated by the pair  $(t_k, v_k)$ , representing the times and fractions of the susceptible population to be vaccinated. In the contingent control time window, again, the term  $(t_k, v_k)$  has variant values. In contrast, in the timeframe of guardian control,  $(t_k, v_k)$  has constant value. The sum of time intervals must not exceed their bounds  $T_{tmp}$  and  $T_{gc}$ . The number of policies,  $N_{cc}$ , in the contingent control can vary from  $\lfloor T_{tmp} / \Delta t_{max} \rfloor$  to  $\lfloor T_{tmp} / \Delta t_{min} \rfloor$ , while the number of guardian control stages,  $N_{gc}$ , is the same as Equation (4). The terms  $v_k$  and  $\Delta \tau_k$  are vaccination proportion and time periods for the entire system time horizon, where the contingent control starts first and proceeds until administrating  $N_{cc}$  vaccination policies; this is followed by the application of  $N_{gc}$  guardian control vaccination policies. For both stages, there are  $N_{cc}$  different contingent control vaccination policies and  $N_{gc}$  constant guardian control vaccination polices to be applied in the duration  $[0, T_{cntrl}]$ . Henceforth, dividing the time limit of each stage  $T_{tmp}$  and  $T_{gc}$  over the possible policies' durations  $\Delta t_{gc}$  and  $\{\Delta t_1, \dots, \Delta t_{cc}\}$  produces the number of policies in each stage  $N_{gc}$  and  $N_{cc}$ , respectively. The total number of policies in the final vaccination campaign is  $N_{Total} = N_{cc} + N_{gc}$ . Following the same condition employed in phase 1, the disease should remain below an acceptable value, where  $(I) \leq i_{tol} \cdot N$  (i.e.,  $i_{tol} = 0.01$ ), to be maintained by applying the selected policy from the guardian control stage repeatedly until  $T_{cntrl}$ . To simplify the system, (R) compartment again is not mentioned, as declared earlier.

### B. Chromosome Representation

The representation of a chromosome is composed of a series of genes. The chromosome is mainly divided into three sub-lists; genes in the first part symbolize the timeframe period in each gene to represent  $\tau_{k+1} - \tau_k = \Delta t_k$ , while the second part of the chromosome indicates the set of vaccination polices, as each gene represents a proportion of the vaccination  $v_k$ . The last part shows the policy chosen from the guardian phase, where  $t_{gc}$  is the time period between two consecutive vaccination policies and  $v_{gc}$  is the proportion to be vaccinated, as shown in Figure 3.

The first two parts of the chromosome belong to the contingent phase. The chromosome is depicted as a sequence of times followed by a sequence of vaccination proportions that will be employed at each abovementioned time period. It is worth noting that the chromosome lengths are of variable size, related to the number of vaccination policies applied over the indicated time horizon. The use of variable-length chromosomes (VLC) is motivated by many works found in the literature, such as [24,25,40–43]; for this work, no mapping from genotype to phenotype is needed, so there is no encoding/decoding process. To serve the purpose of this paper, VLC is utilized without adding an extra burden on the CPU time or memory utilization during the evolutionary processes (i.e., crossover/mutation). The condition applied to the chromosome length is that the number of timeframe periods must be the same as the number of vaccination policies. In other words, for each timeframe, an associate vaccination policy is to be applied.

The lower and upper limits of policies in a chromosome are fixed. We begin by defining a particular limit to the number of policies in a chromosome, say,  $1 \leq k \leq 20$ . However, we do not know exactly how many policies are required. This should also be identified by  $\Delta t_k$  on the fitness function, where  $0 \leq \sum \Delta t_k \leq T$ . So, the chromosome should be able to increase or decrease the number of policies from individual to individual, or from generation to generation.

### 5.2. Non-Dominated Sorting Genetic Algorithm (NSGA)

A new paradigm of multi-objective algorithms has recently been dominating in the literature. These algorithms do not switch multi-objective problems into single-objective ones; instead, they are heuristics oriented towards guiding the multi-objective search. In order to tackle the complex vaccination problem with its contradictory objectives, a compromise-based approach must be applied to find the set of near-optimal solutions. In principle, the evolutionary multi-objective problem produces a set of optimal solutions (known as non-dominated or Pareto-optimal solutions), rather than a single optimal solution.

The motivation for using the multi-objective approach in solving complex problems in optimization is that it does not require special equations, which helps to simplify the problem. The nature of multi-objective optimization allows for a compromise (tradeoff) between some contradictory issues, and then leaving the final decision to the policymaker. In other words, there is no single best solution for all dimensions, but rather there are several candidate solutions [29]. In the current paper, the proposed mathematical model is solved by applying two Pareto-based metaheuristic algorithms: NSGA-II and CENSGA.

#### A. NSGA-II

A non-elitist multi-objective genetic algorithm NSGA (or NSGA-I), proposed by Deb, Agarwal and Meyarivan [44], was one of the first EAs that was able to find multiple Pareto-optimal solutions in one single simulation run. Several drawbacks were noted, such as the high computational complexity  $O(MN^3)$ , lack of elitism and the necessity of a sharing parameter. NSGA serves as a basis of two other metaheuristic algorithms, NSGA-II and CENSGA, whose implementations differ in several aspects. The powerful NSGA-II is one of the most widely implemented algorithms within this category; it is well known for its outstanding enhanced features, such as its fast non-dominating sorting-procedure  $O(MN^2)$ , elitist-preserving mechanism and no-niching parameter. The main idea of NSGA-II is to reproduce a new population from an initial population without losing good solutions, and assure better convergence to the optimal Pareto-optimal front with a good spread of the solutions [44].

The best solutions and Pareto fronts are obtained by prioritizing solutions using non-dominated sorting. Non-dominated sorting considers two parameters:  $N_p$ , which represents the number of solutions dominating a certain solution, and  $S_p$ , which represents the set of solutions derived by a certain solution. The sorting process is a labelling procedure carried out repeatedly, wherein each solution must be ranked with a number, potentially non-unique, that indicates the front it belongs to. For minimization problems, the best solution is assigned to the first front, the second solution is assigned to the second front, and

so on. The ranking of the solutions works as a measure of fitness. The fast non-dominated sorting operator is used to associate a rank to each solution relative to its dominance level. However, within the same rank, a set of solutions is assigned. Thereby, a measure of the density of solutions belonging to the same rank (front) is applied, which is denoted as the crowding distance (CD) [14]. Algorithm 1 outlines a summary of the NSGA-II algorithm [12].

---

**Algorithm 1.** Template for NSGA-II algorithm [12]

---

```

 $R_t = P_t \cup Q_t$  //combine parent  $P_t$  and children population  $Q_t$ 
 $F = \text{fast-non-dominated-sort}(R_t)$  //rank all non-dominated fronts of  $R_t$ 
 $P_{t+1} = \emptyset$  and  $i = 1$ 
until  $|P_{t+1}| + |F_i| \leq N$  //until the new population is filled
     $P_{t+1} = P_{t+1} \cup F_i$  //include  $i$ -th non-dominated front in  $P_{t+1}$ 
    //calculate crowding distance for each solution in  $F_i$  for partial inclusion
    crowding-distance-assignment ( $F_i$ )
     $i = i + 1$ 
    Sort ( $F_i, \prec_n$ ) //sort in ascending order using  $\prec_n$ 
 $P_{t+1} = P_{t+1} \cup F_i[1:(N - |P_{t+1}|)]$  //choose the first  $N$  elements in  $F_i$ 
 $Q_{t+1} = \text{make-new-pop}(P_{t+1})$  //using selection, crossover, and mutation
 $t = t + 1$ 

```

---

The new population pool  $P_{t+1}$  is created by adding solutions from fronts in a consecutive manner, starting from the first front  $F_1$ , until the size of the parenting pool reaches  $N$ . Individuals on all fronts are ordered using a crowding distance that calculates the distance between neighboring solutions. Thereafter, the best fronts are included entirely in NSGA-II, while solutions of the last-accepted front are sorted and selected according to a crowded comparison criterion. The crowded comparison criterion uses a special relation  $\prec_n$  to promote diversity in the solutions within this front, where  $\prec_n$  is defined as follows:

$$\text{if } (i_{\text{rank}} < j_{\text{rank}}) \text{ or } ((i_{\text{rank}} = j_{\text{rank}}) \text{ and } (i_{\text{distance}} > j_{\text{distance}})), \text{ then, SELECT } i$$

In other words, the solutions with lower ranks are preferred when there are differences in the non-dominated ranks. Apart from that, if both solutions are on the same front, then the larger crowding distance (i.e., solutions from less dense regions) is preferred. This forms the population  $P_{t+1}$  of size  $N$ . At this point, the parent population  $P_{t+1}$  proceeds for selection, crossover, and mutation to construct a new population  $Q_{t+1}$  of size  $N$  [12]. In our work, the bounded Simulated Binary Crossover (SBX) [45] and bounded Polynomial Mutation [46] are used as recombination operators, where these are the same GA operators used in [6].

## B. CENSGA

In this paper, we present an extended version of NSGA-II that is denoted by Controlled NSGA-II or CENSGA. CENSGA was first proposed by Deb and Goel [12], with the main differences related to the selection strategy. In particular, CENSGA gives all fronts the ability to participate in the selection process through a geometric distribution with various priorities that reflect the front level. Figure 4 shows that the reproduction of the new population under CENSGA is more varied than under NSGA-II.

The two-fold affirmation of elitist solutions in NSGA-II will cause a rapid exclusion of solutions belonging to non-elitist fronts. Although the CD operator will preserve diversity along the current non-dominated front, lateral diversity will be absent. Basically, under such conditions, the search may wane due to the lack of diversity in certain decision variables. Thus, the ability to direct the search towards better regions of optimality may be paralyzed. Therefore, in order to prevent premature convergence, a search algorithm more effectively adopts diversity in both directions—along the Pareto-optimal front and lateral to the Pareto-optimal front [12], as presented in Figure 5.

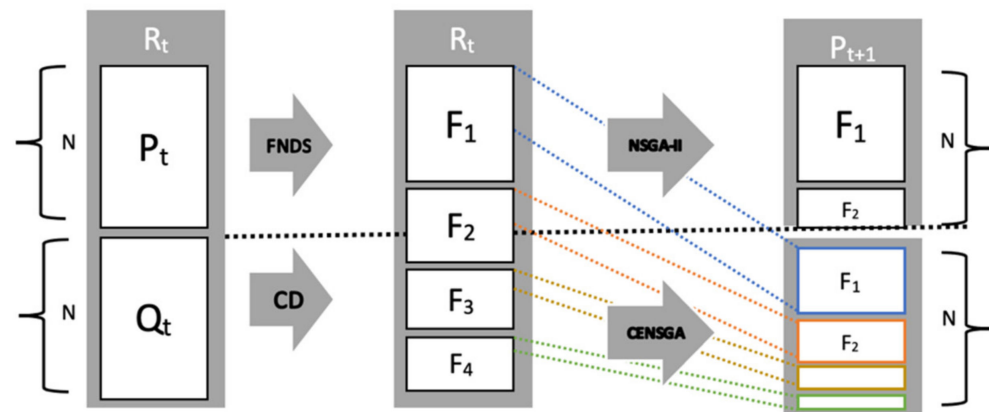


Figure 4. Selection strategy of CENSGA.

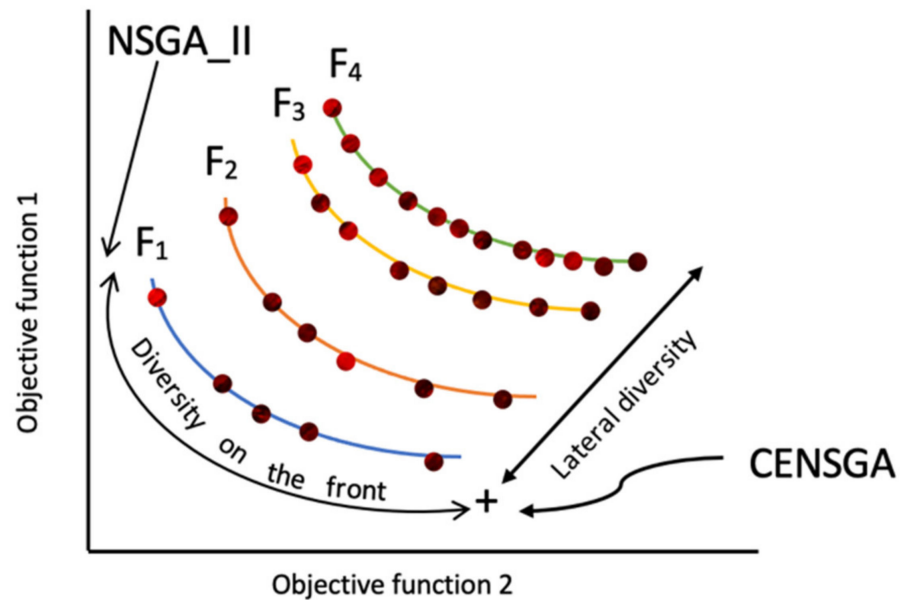


Figure 5. Controlled elitism procedure is illustrated.

In CENSGA, the selection operator is modified to encourage more diversity by allowing all fronts to participate in the selection strategy. The participation of fronts is controlled by a geometric distribution that prioritizes them, ensuring that better fronts have a greater influence in shaping the next generation. This process is calculated by Equation (8):

$$n_i = r \cdot n_{i-1} \tag{8}$$

where  $n_i$  represents the number of individuals in the  $i$ th front and  $r$  ( $<1$ ) identifies the reduction rate. In a population of size  $N$ , let  $k$  be the number of non-dominated fronts; then, the maximum number of individuals allowed in each front  $f_i \in (1, 2, \dots, k)$  is computed as follows:

$$n_i = N \cdot r^{i-1} \frac{1-r}{1-r^k} \tag{9}$$

At front  $f_i$ , the selection of  $n_i$  individuals is performed by the CD operator. The geometric distribution ensures an exponential reduction in the number of solutions over the fronts [14,47]. Algorithm 2 shows the pseudo-code for the controlled selection, while Figure A1 presents a flowchart of the CENSGA algorithm.

**Algorithm 2.** Pseudo-code for CENSGA Controlled geometric selection

---

```

 $R_t = P_t \cup Q_t$  //combine parent  $P_t$  and children population  $Q_t$ 
 $F = \text{fast-non-dominated-sort}(R_t)$  //rank all non-dominated fronts of  $R_t$ 
 $P_{t+1} = \emptyset$  and  $i = 1$ 
repeat
     $n_i = N \cdot r^{i-1} (1-r) / (1-r^k)$  //Geometric distribution to decide
                                                number of individuals at front  $i$ 
    crowding-distance-assignment ( $F_i$ ) //calculate crowding distance for each solution
in
     $F_i$  for partial inclusion
    Sort( $F_i, \prec_n$ ) //sort in ascending order using  $\prec_n$ 
     $P_{t+1} = P_{t+1} \cup F_i [1:n_i]$  //include  $n_i$  non-dominated front in
                                                 $P_{t+1}$  according to Geometric distribution
     $i = i + 1$ 
until  $|P_{t+1}| + |F_i [1:n_i]| \leq N$  //until the new population is filled
 $Q_{t+1} = \text{make-new-pop}(P_{t+1})$  //using selection, crossover, and mutation
 $t = t + 1$ 

```

---

### C. Local search via Gaussian approximation.

Deterministic search methods were proven to find the local optimal solution with fast and precise convergence properties. A promising means to enhance the local search beyond local optimality is by combining evolutionary algorithms and classical deterministic search methods. Our work is inspired by a local search scheme proposed in [6] that boosts the convergence and improves the quality of solutions. This hybrid method selects solutions from the non-dominated front (recombined population) to feed the local search's initial population. The Gaussian approximation with scalarized weighted sums [29] is used as a function to generate newly approximated Pareto-optimal solutions, which may yield new opportunities to search areas closer to the true optimal front.

The hybrid scheme embeds the local search in the canonical NSGA-II and CENSGA algorithms, such that the process of the local search is triggered every 20 generations. The local search has the following fundamental actions: (1) A set of ( $r$ ) solutions is chosen at random from the existing non-dominated front. (2) Around each solution ( $r$ ) chosen in the former step, a set of ( $m$ ) new points is created autonomously, by Gaussian distribution, with standard deviations equal to 0.01 times the size of the search space for each decision dimension. (3) The fitness value is evaluated on the new points to associate their objective values. (4) The new  $r \times m$  solutions are introduced into the current population.

The setup of the experiments conducted here is as follows: ( $r$ ) is set to hold four randomly chosen solutions selected from the existing non-dominated front, with  $m = 2s$ , and  $s = 2 \cdot n + 1$ , where ( $n$ ) is the search space dimension of the problem. The number of newly produced solutions for each set of approximated functions has been set as  $p = N/2$ , where  $N$  is the population size of the evolutionary algorithm. In other words, in each iteration, when the local search is triggered,  $2N$  new solutions are produced; then, the selection operator is performed over  $4N$  solutions, with  $2N$  solutions coming from the current iteration of the optimization process and  $2N$  coming from the local search.

### D. Evolutionary algorithm using a hash table

In order to maintain a unique set of solutions and most effectively utilize the processor time, a hash table was incorporated into the evolutionary algorithm, as proposed in [6]. A good data structure and a good hashing function has a mean search time computationally equivalent to  $O(1)$  for an element in the array [48].

The current research benefits from a hash table in storing the non-repetitive generated solutions during the evolutionary iterations in the optimization process. When a new offspring is obtained by a genetic operator, it passes through a verification process to acknowledge that there is no comparable member on the hash relation. This step assures

there is no repetition among the hash relation members. When an offspring matches a member stored in the hash table, a random vector with Gaussian distribution (zero-mean and standard deviation equal to 1.0 percent of the search space dimension) is added to the offspring coordinates, generating a new offspring around the original member. In this way, the hash table encourages diversity, and only new solutions are evaluated.

During the optimization run, the algorithm is executed for several generations, and on each iteration, the hash table is updated accordingly. Hence, the relation of solutions is set with the initial population and expands in each iteration of the optimization process. In the final iteration, non-dominated sorting is applied over all stored solutions during the optimization process. The return result is a set with all non-dominated solutions that were involved in all executions. An outline of the CENSGA algorithm is presented in Algorithm 3.

---

**Algorithm 3.** CENSGA Outline Algorithm

---

```

P1 ← InitialPopulation() //P1 is the initial population generations
[H, P1] ← AnalyzeInsertHashTable(H, P1) //H is the hash table
[H, P1] ← EvaluatePopulation(H, P1)
NDS1 ← NonDominatedSorting(P1) //NDS contains First Pareto front of P1
NDS1 ← CrowdingDistance(NDS1)
for t from 1 to t step 1 do //t is the number of generations
  Qt ← GeometricSelectionAndRecombination(Pt)
  Qt ← Crossover&Mutation(Qt)
  [H, Qt] ← AnalyzeInsertHashTable(H, Qt)
  if it is desired to execute local search then
    L ← LocalSearch(NDSt, H)
    Qt ← Qt ∪ L
  end if
  [H, Qt] ← EvaluatePopulation(H, Qt)
  Rt ← Pt ∪ Qt
  NDSt ← NonDominatedSorting(Rt)
  NDSt ← CrowdingDistance(NDSt)
  Pt+1 ← FillNondominated(GeometricSelection(NDSt))
end for
PF ← NonDominatedSorting(H) //Final pareto front PF

```

---

## 6. Parameter Tuning

Different approaches can be utilized to tune the parameters of an algorithm [49–51]. In this work, the Taguchi method is applied to calibrate the parameters of the two metaheuristic algorithms tested. The Taguchi method is a robust and powerful tool for optimizing the performance settings of algorithms [52]. The design of the Taguchi test involves the concept of orthogonal arrays (OA) used to study the effects of different permutations of decision variables or factors on many levels with less trials, contrary to performing all analyses, as in the full factorial design [51]. In Taguchi designs, the signal-to-noise (SNR) ratio is used to find the optimal level of control factors and to reduce the effect of noise factors. Control factors, in our case, are the algorithm's parameters that can be controlled. Noise factors cannot be controlled while the algorithm is executed, but can be controlled during experimentation. Therefore, in a Taguchi-designed experiment, the noise factors can be manipulated to encourage underlying variability to occur to identify optimal parameters settings that make the algorithm robust, or at least resistant to variation from these noise factors. Thus, the signal-to-noise (SNR) ratio is employed. Furthermore, the expected outcome (response) is minimization, so a smaller-the-better type of response is applied. Equation (10) represents the SNR ratio as follows [53]:

$$SNR = -10 \times \log(S(G^2)/n) \quad (10)$$



where  $G$  is the responses used for the respective parameter levels and  $n$  is the number of responses in the parameter levels.

The parameter calibration test is executed by the Orthogonal Arrays (OA) (i.e., Taguchi representation) L9 ( $3^4$ ) and L27 ( $3^5$ ); that is, 9 and 27 test runs are designed from four and five parameters, with three levels of potential parameter values. Table 2 shows the parameters of NSGA-II and CENSGA, and the levels defined for them. Using Minitab software, for NSGA-II and CENSGA, the effect plots for the SNR ratio are computed.

**Table 2.** The levels defined for the parameters of NSGA-II and CENSGA.

Algorithm	Parameters	Parameters Levels		
		Level 1	Level 2	Level 3
NSGA-II	Population size (A)	25	50	70
	Maximum generation (B)	30	50	100
	Crossover percentage (C)	0.6	0.75	1
	Mutation percentage (D)	0.1	0.2	0.4
CENSGA	Population size (A)	25	50	70
	Maximum generation (B)	30	50	100
	Geometric distribution (C)	0.1	0.5	0.9
	Crossover percentage (D)	0.6	0.75	1
	Mutation percentage (E)	0.1	0.2	0.4

The objective function values are computed by implementing the algorithms for the two-staged Pulse vaccination problem. For each combination of parameters, the bi-objective functions of the model,  $z_1$  and  $z_2$ , are considered jointly, and two objective values are obtained. Hence, for each objective function, the mean of the population's values obtained for the problem can be calculated. This produces two mean values that are converted to a single objective function value using the weighted-sum approach [29].

$$Z = w_1 \times Z_1 + w_2 \times Z_2 \quad (11)$$

where the weight parameters  $w_1$  and  $w_2$  indicate the contribution of the objectives to the combined objective value; in this work, the values are set to 0.5, indicating equally significant objectives. Figures 6 and 7 depict the SNR ratio of the Taguchi test runs for NSGA-II and CENSGA, respectively. In NSGA-II, the SNR ratio is applied twice, since the first round indicates a significant value for the population size among the other parameters only. Therefore, a second SNR round is applied by stabilizing the first parameter with the best value obtained from the first round, using the same levels for the other parameters employed when using representation L9 ( $3^3$ ). The best calibrations of the parameters determined by the Taguchi method are tabulated in Table 3.

**Table 3.** Optimal values for the parameters of NSGA-II and CENSGA.

Algorithm	Parameters	Optimal Value
NSGA-II	Population size (A)	70
	Maximum generation (B)	50
	Crossover percentage (C)	1
	Mutation percentage (D)	0.2
CENSGA	Population size (A)	70
	Maximum generation (B)	100
	Geometric distribution (C)	0.9
	Crossover percentage (D)	1
	Mutation percentage (E)	0.1

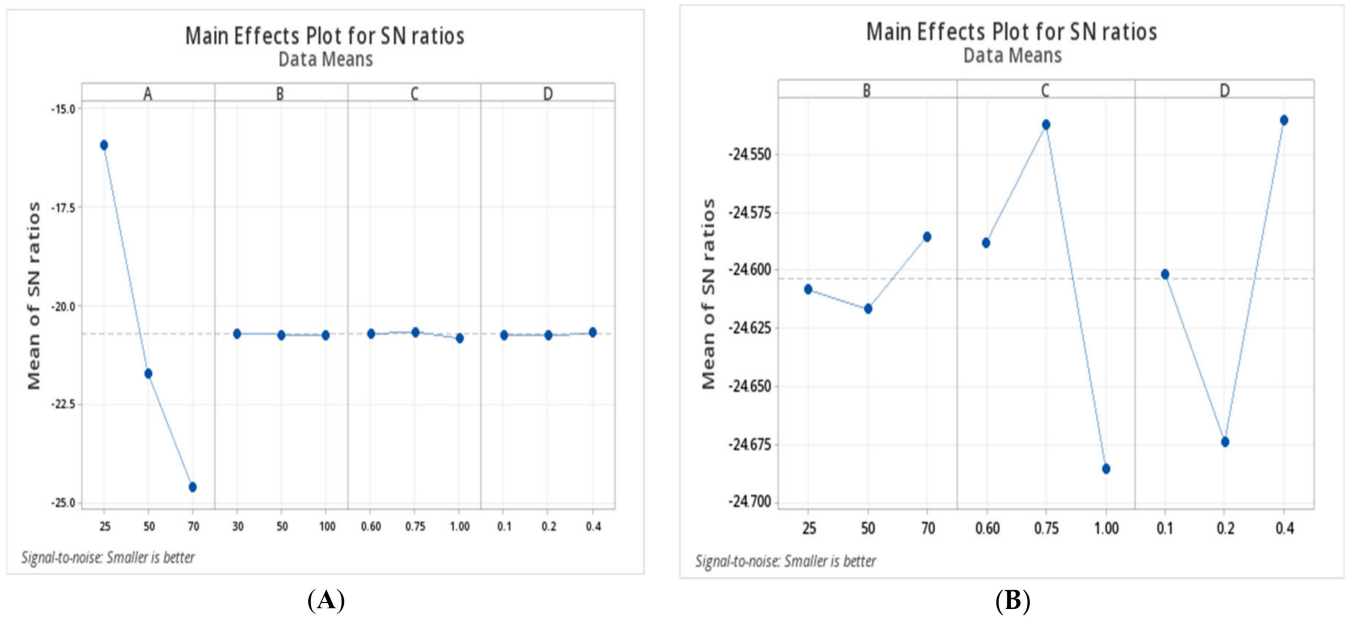


Figure 6. Main effects plot for the SN ratio of NSGA-II in (A) round 1 and (B) round 2.

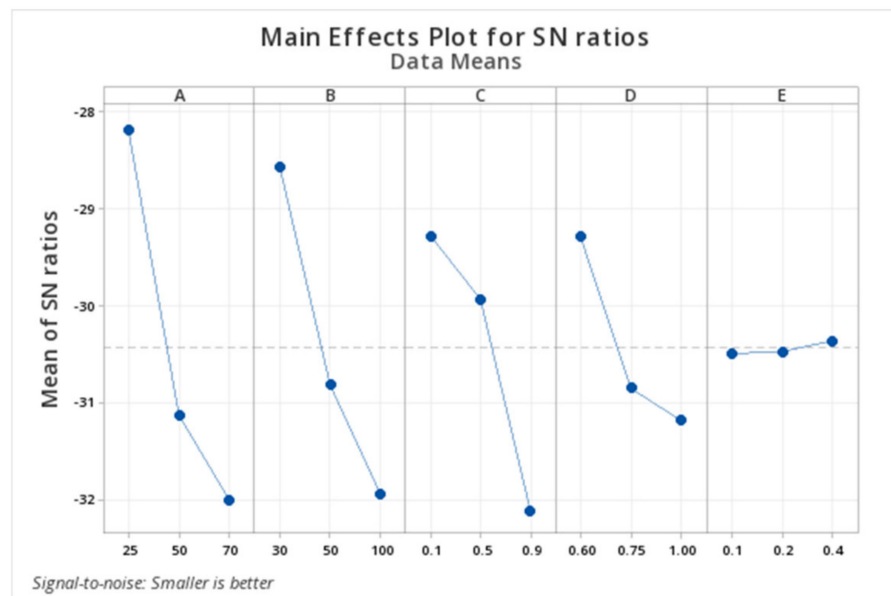
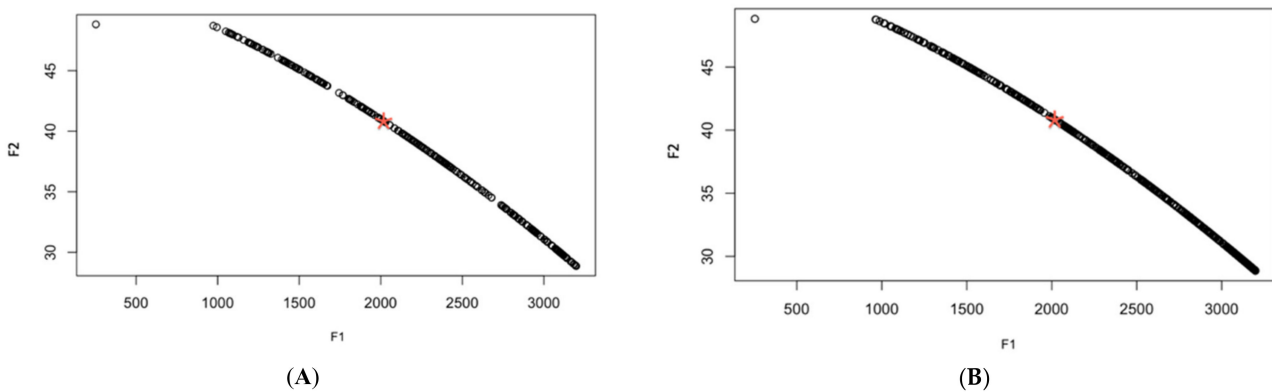


Figure 7. Main effects plot for SN ratio of CENSGA.

### 7. Results

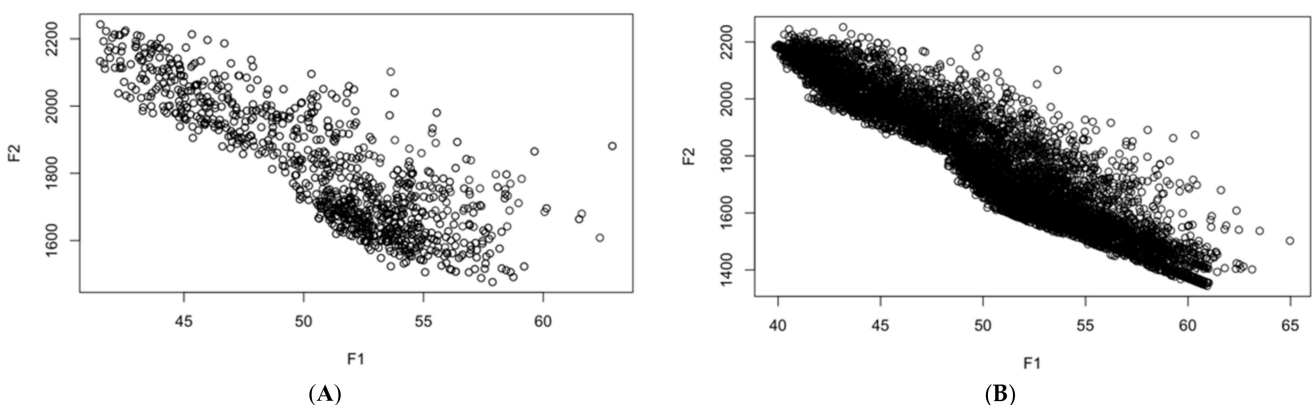
After the execution of the multi-objective optimization algorithms for the guardian phase of the problem formulated in Equation (4), a set of non-dominated guardian control solutions were determined, with 597 and 1132 solutions for NSGA-II and CENSGA, respectively. All these solutions verify the constraints and belong to the non-dominated Pareto front. In this phase, the evolutionary algorithms evaluated 6000 and 25,248 different solutions for NSGA-II and CENSGA, respectively. Figure 8 shows the non-dominated Pareto front of the guardian phase of the problem.



**Figure 8.** Non-dominated Pareto front of the guardian phase of the problem. Where, Black circles are solutions belong to PF, while red star indicates the selected solution  $\Delta t_{gc} = 8.3335$  and  $v_{gc} = 0.8752$ . (A) NSGA-II, (B) CENSGA.

The outcome of the guardian phase is passed to the contingent phase as a fixed policy, since there is a recurrent Pulse guardian policy that will be applied to control the infection rate. Consequently, the guardian control policy was selected from the optimal non-dominated solution of the guardian phase, which is marked with a red symbol in Figure 8. This corresponds to the guardian control policy  $\Delta t_{gc} = 8.3335$  and  $v_{gc} = 0.8752$ , which were included in Equation (6).

All solutions generated by the evolutionary algorithms with the local search operator based on a hash table are visualized in Figure 9, where 77,000 solutions for NSGA-II and 637,560 for CENSGA are evaluated in all executions of the contingent phase. In general, the solutions obtained by the two algorithms belong to the same exploration area. Conversely, CENSGA can exploit more due to the introduction of Controlled elitism procedure; the search algorithm effectively obtains diversity in both directions, i.e., the Pareto-optimal front and lateral to the Pareto-optimal front. The Pareto front of the contingent phase of the optimization process is represented in Figure 10. All those solutions verify all constraints. The red star marks the selected Pareto-optimal solution for both fronts from the optimal non-dominated solution set. The final non-dominated set of complete controls shows a higher number of smooth, better spaced and more variable solutions on CENSGA’s Pareto front compared to NSGA-II’s Pareto front.



**Figure 9.** Image set of all obtained solutions considering all executions of the evolutionary algorithms for the contingent phase presented in Equation (6). (A) NSGA-II, (B) CENSGA.

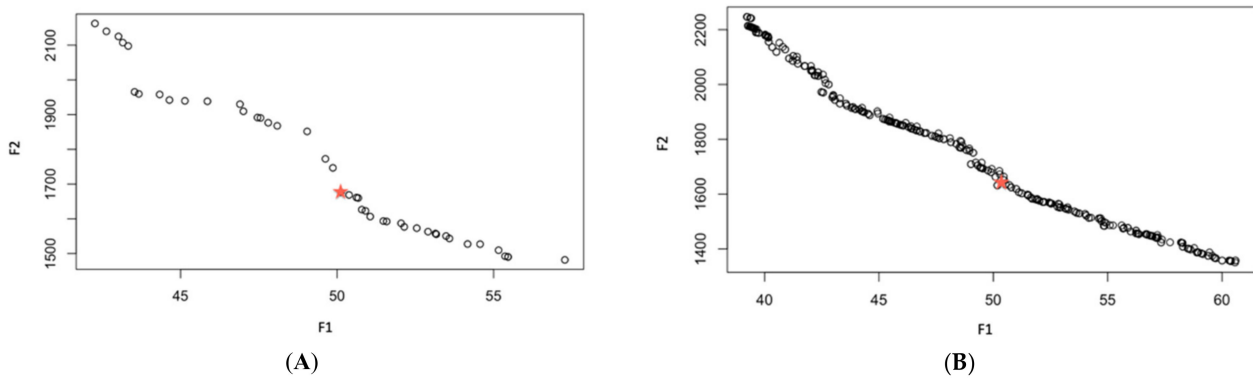


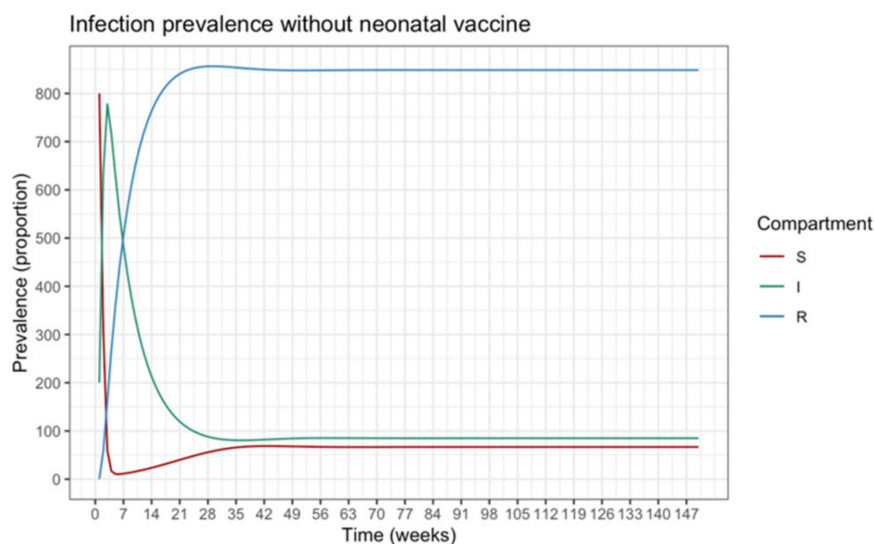
Figure 10. Non-dominated Pareto front of the contingent phase problem. (A) NSGA-II, (B) CENSGA.

A fast non-dominated sorting procedure was executed using the objective functions of the final hash table. A partial set of the sorted non-dominated set of robust complete control policies is presented in Table 4. This table shows the top 30 solutions’ objective function values. Two probabilities are defined:  $P_C$  indicates the disease is “under control” after the end of the contingent phase, calculated as the ratio of realizations in which the number of infected individuals approaches  $0.01N$  or below; and the probability of disease eradication,  $P_{RE}$ , is defined as the ratio of realizations in which  $R_{EF}$  is under one within the simulation time horizon.

Table 4. Top 30 non-dominated solutions.

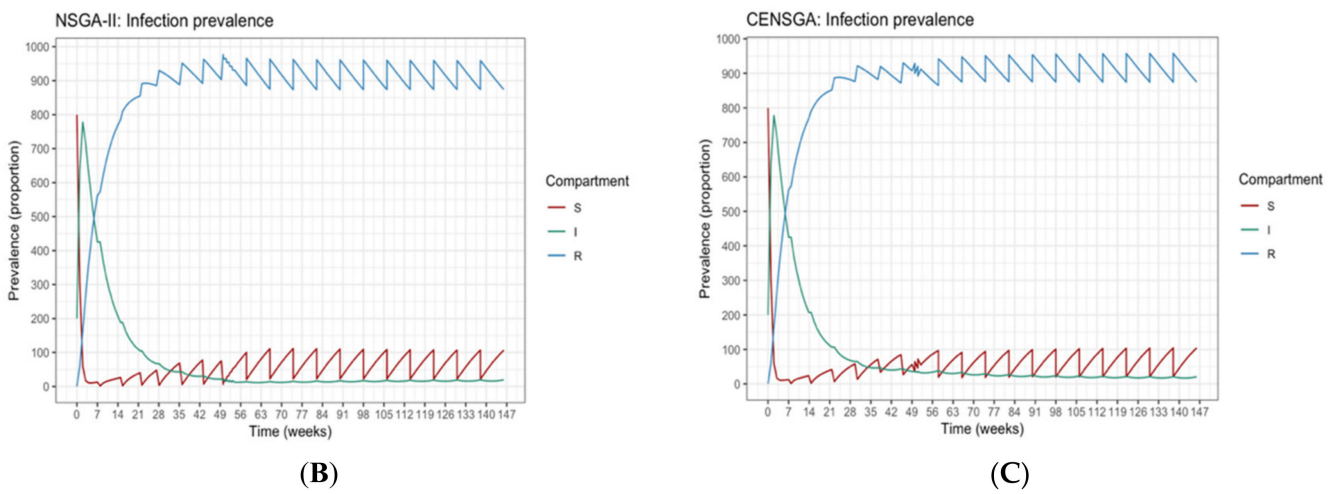
Solution #	NSGA-II				CENSGA			
	F1	F2	$P_C$	$P_{RE}$	F1	F2	$P_C$	$P_{RE}$
1	57.27546	1481.784	0.03	0.58	40.17971	2155.631	0.93	0.56
2	42.26528	2162.272	0.91	0.56	39.22617	2246.802	0.96	0.57
3	42.48387	1972.431	0.91	0.56	42.48387	1972.431	0.51	0.59
4	43.32805	2097.467	0.90	0.61	49.02240	1709.273	0.91	0.61
5	43.53146	1965.415	0.50	0.58	42.34167	2031.065	0.72	0.58
6	49.04402	1851.618	0.17	0.57	50.17826	1632.085	0.60	0.60
7	49.62623	1772.659	0.08	0.58	48.95636	1757.900	0.92	0.61
8	<b>50.12384</b>	<b>1676.411</b>	<b>0.58</b>	<b>0.60</b>	57.33833	1423.545	0.02	0.54
9	49.86542	1746.725	0.88	0.61	55.67879	1474.348	0.05	0.59
10	55.46100	1490.285	0.07	0.58	<b>50.09297</b>	<b>1663.320</b>	<b>0.74</b>	<b>0.61</b>
11	46.89765	1929.952	0.62	0.59	40.51751	2118.797	0.91	0.51
12	49.62623	1772.659	0.08	0.58	42.56096	1970.538	0.62	0.60
13	48.08837	1867.685	0.45	0.58	39.26545	2214.084	0.95	0.57
14	44.33212	1957.920	0.58	0.58	39.24922	2246.283	0.96	0.57
15	55.15776	1509.711	0.09	0.58	54.85511	1483.855	0.09	0.58
16	50.78942	1626.521	0.44	0.60	41.07164	2095.011	0.91	0.55
17	45.85773	1938.358	0.72	0.59	50.78363	1624.502	0.46	0.60
18	42.63134	2139.678	0.92	0.56	44.59067	1887.626	0.40	0.61
19	47.00748	1909.577	0.55	0.59	40.33498	2136.160	0.92	0.52
20	50.67661	1660.034	0.49	0.60	54.61229	1511.137	0.12	0.57
21	45.85773	1938.358	0.72	0.59	45.18799	1874.321	0.33	0.59
22	44.64455	1941.996	0.45	0.58	42.96260	1953.169	0.57	0.61
23	54.56721	1527.107	0.12	0.58	54.17620	1513.515	0.13	0.58
24	43.01730	2124.942	0.90	0.57	40.33498	2136.160	0.92	0.52
25	51.05753	1606.500	0.30	0.59	43.29324	1930.275	0.54	0.59
26	54.16909	1527.488	0.14	0.59	49.90359	1679.994	0.64	0.59
27	43.66610	1959.621	0.46	0.58	54.73271	1501.457	0.07	0.58
28	45.13841	1939.580	0.27	0.56	48.38427	1783.314	0.15	0.61
29	53.58785	1543.445	0.18	0.60	41.45457	2076.025	0.85	0.60
30	52.04441	1587.148	0.30	0.60	51.00140	1618.919	0.40	0.59
Average	<b>48.393984</b>	<b>1807.51157</b>	<b>0.46</b>	<b>0.58</b>	<b>46.4923847</b>	<b>1847.38473</b>	<b>0.58</b>	<b>0.58</b>

The performance indices of the top 30 solutions of NSGA-II and CENSGA are indicated in Table 4. The mean values of  $P_C$  and  $P_{RE}$  presented in Table 4 and Figures 11–13 are now analyzed for the sake of comparison. The data are analyzed using the indicated two probabilities  $P_C$  and  $P_{RE}$ . By definition, multi-objective problems are characterized by the fact that an improvement in one objective usually causes a deterioration in the other objective, which may lead to a biased judgment if we build our comparison based only on objective values. Thus, we selected two candidate solutions from both Pareto fronts by marking the midpoints in Figure 10 with a red star (i.e., solution 8 for NSGA-II and solution 10 for CENSGA). The candidate solutions represent average and non-extreme points among their relative fronts. It is noticeable that, on average, CENSGA outperforms NSGA-II with respect to objective one value, and in the performance probability  $P_C$ , while in the probability  $P_{RE}$ , both algorithms perform equally. Furthermore, the mean probability of disease eradication is increased from 46% to 58% (an increase of almost 12%), while the probability  $P_{RE}$  holds its position. Besides this, we can observe a slight geometric closeness among the candidate solutions, with an increase in the probability of disease eradication from 58% to 74% (an increase of 16%); however, the probability of  $P_{RE}$  records a limited improvement, by 1%, in favor of the CENSGA policy. This means that the candidate CENSGA solution tends to be both financially cheaper and more effective in disease control. This can be explained by the “time is money policy”, which reflects that the correct determination of campaign timing is a key aspect of vaccination, which is reflected in cost saving. It is worth mentioning that both algorithms perform well in the selected solutions. A quick glance at Figure 12 indicates that the CENSGA solution suggests almost the same number of vaccination campaigns (minus one) as the NSGA-II control, while decreasing the proportion of the vaccination ratio (refer to Figure 12). This enhances the considered time window in which the number of those susceptible becomes lower and the number of those recovered becomes higher—due to the combined effect of the vaccination and the correct initiation of the campaigns. This is reflected more obviously in the number of realizations in which the disease is under control and progressing to eradication. The SIR behavior of these solutions is visualized in Figure 11. The mean number of vaccinated individuals in each campaign is presented in Figure 12, while Figure 13 outlines the difference in mean performance of  $P_C$  and  $P_{RE}$ .

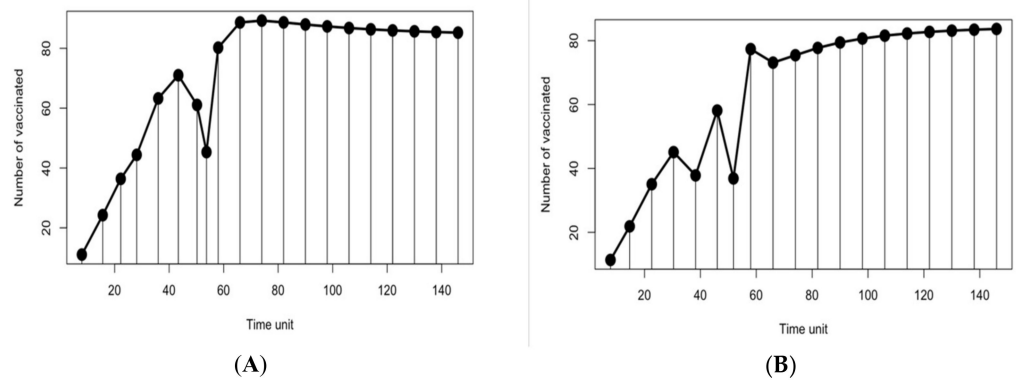


(A)

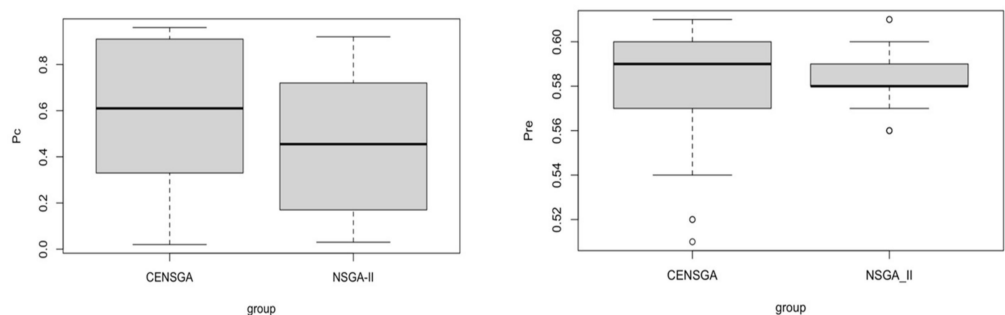
Figure 11. Cont.



**Figure 11.** SIR behavior under different situation. (A) SIR general behavior. (B) Behavior of the selected solution from NSGA-II. (C) Behavior of the selected solution from CENSGA.



**Figure 12.** Mean number of vaccinations for (A) solution 8 from the NSGA-II non-dominated set and (B) solution 10 from the CENSGA non-dominated set.



**Figure 13.** Mean values of  $P_C$  and  $P_{RE}$ .

It is worth mentioning that the candidate solutions may have better options, such as solutions 4 and 9 for NSGA-II and solutions 4 and 7 for CENSGA. However, all these solutions are rejected, even though they show better  $P_C$  and  $P_{RE}$  values, as they all involve costly vaccination campaigns. The same narrative is applied to other similar solutions as well.

### 8. Efficiency Performance Measures

The quality indicators (QI) for multi-objective optimization involve considerably large numbers of performance measures. These measurements have been proposed to

approximate the quality of the produced Pareto front. In this work, the algorithms are coded using R Studio 2022.02.2 (Build 485) and implemented on a PC using MacOS Big Sur, 2.40 GHz Quad-Core Intel Core i5, RAM 16 GB 2133 MHz LPDDR3. To evaluate the algorithms, we applied a set of performance indicators that fall under four principal classes: cardinality, convergence, distribution and spread. In practice, when the real Pareto front is unknown, the reference set often consists of the non-dominated solutions of the collections of all solutions produced during the search [54]. For instance, we combined the Pareto fronts of NSGA-II and CENSGA, and then applied fast non-dominated sorting, ranking and crowd distance functions in the collection to produce the reference set.

1. Error ratio (ER)

This measure of cardinality is given by the following formula:

$$ER(P) = \frac{1}{|P|} \sum_{a \in P} e_a \tag{12}$$

where

$$e_a \begin{cases} 0 & \text{if point } a \text{ belongs to } PF \\ 1 & \text{otherwise} \end{cases}$$

A set of non-dominated points  $\in (P)$  within the reference set ( $PF$ ) will have an error ratio of 0; otherwise, the error ratio will be assigned as 1 [54].

2. Generational distance (GD)

This indicator of convergence is given by the following formula:

$$GD(P, PF) = \frac{1}{|P|} \left( \sum_{a \in P} \min_{r \in PF} \|F(a) - F(r)\|^2 \right)^{\frac{1}{2}} \tag{13}$$

$GD$  measures the distance between a given Pareto front ( $P$ ) with the reference set ( $PF$ ), where  $\min_{r \in PF} \|F(a) - F(r)\|$  denotes the minimum Euclidean distance between point ( $a$ ) and the points  $\in (PF)$ . This indicator measures the distance between the Pareto front of a particular algorithm and the reference set. The lower the outcome of  $GD$ , the better the performance is [55].

3.  $\epsilon$ -indicator ( $\epsilon$ -IQ)

This indicator of convergence and spread is given by the following formula:

$$\epsilon_X(P, PF) = \max_{pf \in PF} \min_{p \in P} \max_{j \in \{1..m\}} \frac{pf_j}{p_j} \tag{14}$$

The  $\epsilon$ -indicator is to be minimized, where  $\epsilon_X(P_1, PF) < \epsilon_X(P_2, PF)$  indicates that the set  $P_1$  dominates  $P_2$ . It scales the gap of a particular set to the true Pareto front, represented here by the reference set [54].

4. Hypervolume metric (HV)

This indicator of convergence and distribution is given by the following formula:

$$HV(P, PF) = \lambda \cup_{a \in P, a \leq r} [a; r] \tag{15}$$

where  $\lambda$  denotes the Lebesgue measure [56]. In other words, a measure of the union of boxes. Given Pareto front  $P \subset R^m$  with a reference point  $r \in PF$ , where  $[a; r] = \{s \in R^m \mid a \leq s \text{ and } s \leq r\}$  denotes the box bounded below by  $a \in P$  and above by  $r \in PF$ .  $HV$  favors a higher quality value for a set of given solutions. This can be interpreted as implying that the bigger the volume of the union of the hypercubes determined by each of the Pareto front solutions and the reference set points, the better it is [54].

### 8.1. Analysis and Comparison of Results

Besides the previous case study, a quality performance measurement was performed to evaluate the effectiveness of the proposed optimization algorithm. Model parameters were taken from the literature and are representative of various real estimated influenza scenarios. For testing, a set of different scenarios are suggested based on real epidemic situations. Table A1 characterizes the suggested epidemics with their origin sources. A total of nine computational experiments were planned and executed, considering different SIR parameters settings. For each pair of  $\gamma \in \{1/5, 1/7, 1/9\}$  and  $\beta \in \{1.18, 2.36, 4.72\}$ , NSGA-II and CENSGA were run independently on the last phase of the optimization problem. The parameters used for the optimization algorithms are set similarly. The number of generations was set to 50, the population size to 40, the probability of crossover to 1, the probability of mutation to 0.2 and the distribution index for crossover and mutation to 10. The final quality performance measures are presented in Table 5.

**Table 5.** Nine computational experiments and their associated QI values.

#	Experiments	NSGA-II				CENSGA			
		ER	GD	$\epsilon$ -IQ	HV	ER	GD	$\epsilon$ -IQ	HV
1	$\gamma = 1/5, \beta = 1.18$	0.706	2.368	55.120	0.714	0.326	0	0.018	0.999
2	$\gamma = 1/5, \beta = 2.36$	0.679	1.576	79.512	0.307	0.344	0.453	0.362	0.995
3	$\gamma = 1/5, \beta = 4.72$	0.676	1.0344	80.419	0.31098	0.399	0.239	0.318	0.998
4	$\gamma = 1/7, \beta = 1.18$	0.711	2.633	125.554	0.632	0.353	0.003	0.1195	0.99985
5	$\gamma = 1/7, \beta = 2.36$	0.727	2.581	67.080	0.926	0.305	0.929	1.094	0.992
6	$\gamma = 1/7, \beta = 4.72$	0.742	1.064	41.447	0.271	0.279	0	0	1
7	$\gamma = 1/9, \beta = 1.18$	0.688	1.946	49.832	0.762	0.365	0.845	1.384	0.995
8	$\gamma = 1/9, \beta = 2.36$	0.656	2.108	55.425	0.587	0.4095	0.099	0	0.9992
9	$\gamma = 1/9, \beta = 4.72$	0.813	0.444	25.257	0.474	0.187	0	0	1

The collection of values tabulated in Table 5 indicate the performances of both algorithms in all experiments. In general, CENSGA performed better than NSGA-II in all IQs under different scenarios. For instance, a higher error ratio means the less representative an algorithm is of the reference set. In all experiments, CENSGA has a better representation on the reference set than NSGA-II. The same performance is shown by generational distance, where, in this IQ, it is shown that CENSGA tends to be more convergent to the reference set than NSGA-II, particularly in scenarios 1, 6 and 9, in which all CENSGA points are included in the reference set. The convergence and space IQ (i.e.,  $\epsilon$ -indicator) affirm the findings of the previous two IQs, especially in scenarios 6, 8 and 9, where all CENSGA points are well-spaced over the front. Finally, the distribution of the points on the front was greater in CENSGA in comparison to NSGA-II, which means that CENSGA performs well in terms of the hypervolume indicator in all experiments; in scenarios 2, 3, 6 and 9, the differences are remarkable.

### 8.2. Statistical Analysis

A non-parametric test called the Wilcoxon Rank Sum test was applied to analyze the results statistically [57]. The results for the non-parametric Wilcoxon Rank Sum test for the four metrics are given in Table 6. In this statistical test, the null hypothesis ( $H_0$ ) is defined as “The difference between the two metrics medians are equal”, while the alternative hypothesis is given by “At least one median is different”. The hypothesis is analyzed with confidence 0.95. This means that, if the statistical test returns a  $p$ -value less than 0.05, then the metrics may be considered significantly different.



**Table 6.** *p*-values returned by the Wilcoxon Rank Sum test.

<i>p</i> -Value	<i>ER</i>	<i>GD</i>	$\varepsilon$ - <i>IQ</i>	<i>HV</i>
	0.0004122948	0.001060576	0.000401039	0.0004094601

#### Statistical Test

According to the statistical test results reported in Table 6, the *p*-value of the Wilcoxon Rank Sum non-parametric test for all performance metrics is less than 0.05; thus, H<sub>0</sub> is rejected for all performance measures. In this way, for each IQ measure, the nine experiment cases were compared using the non-parametric Wilcoxon Rank Sum test. For each metric, the indicator values are presented in Figure A2. Statistical differences were detected in all mean experiments according to the *p*-values presented in Table 6. Therefore, it can be concluded that the proposed optimization engine CENSGA with the local search and hash table was outstanding in all mean experiments for diseases with diverse dynamical parameters.

## 9. Conclusions

This paper has presented a multi-objective optimization methodology for allocating a set of vaccination control policies over a specified time horizon. The goal is to minimize the volume of the infected population and the cost of vaccination campaigns. We used an SIR model to simulate an epidemic in order to synthesize a two-phased control system: the guardian control and the contingent control. For each control policy, the time interval between vaccination policies and the proportion to be vaccinated are considered as decision variables. In this way, a high infection curve is mitigated within a cost-wise budget. To solve the synthesis problem, a non-dominated metaheuristic algorithm CENSGA is designed, which provides fast, elitist and efficient solutions. For better convergence, the CENSGA was incorporated with local searches, and a hash table was embedded to hold non-dominated solutions from every generation in order to avoid the repetition or unnecessary reevaluation of old solutions. A final set of Pareto-optimal solutions of the complete vaccination campaign was determined via a multi-objective algorithm, indicating multiple trade-offs between public health and cost units. Thus, the proposed approach delivers alternative robust solutions for vaccination campaigns to the policymaker. The statistical analysis of the proposed approach indicates the significant applicability of the proposed CENSGA compared to the well-known NSGA-II.

The proposed approach of synthesizing a set of optimal vaccination policies via two phases—the guardian and the contingent controls—and the robust dominance approach can be applied in other situations, using models that are more detailed and realistic. In the future, a range of potential extensions can be added, such as an age structure, a spatial structure, seasonality, modeling the epidemic with SEIR and performing a population-based local search, in addition to considering real-life applications.

**Author Contributions:** Conceptualization, A.K.A. and M.H.; methodology, A.K.A.; software, A.K.A.; validation, A.K.A.; formal analysis, A.K.A.; investigation, A.K.A.; resources, A.K.A.; writing—original draft preparation, A.K.A.; writing—review and editing, M.H.; visualization, A.K.A.; supervision, M.H.; project administration, M.H.; funding acquisition, A.K.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the “Research Center of College of Computer and Information Sciences”, Deanship of Scientific Research, King Saud University.

**Data Availability Statement:** Not applicable.

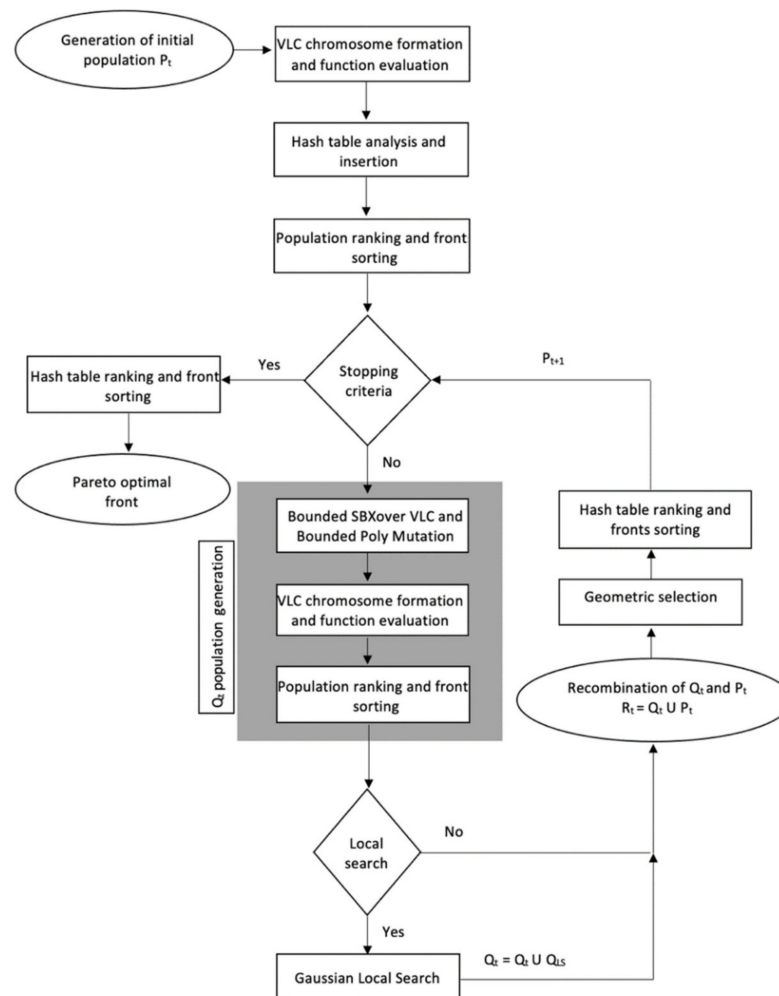
**Acknowledgments:** This research project was supported by a grant from the “Research Center of College of Computer and Information Sciences”, Deanship of Scientific Research, King Saud University.

**Conflicts of Interest:** The authors declare no conflict of interest.

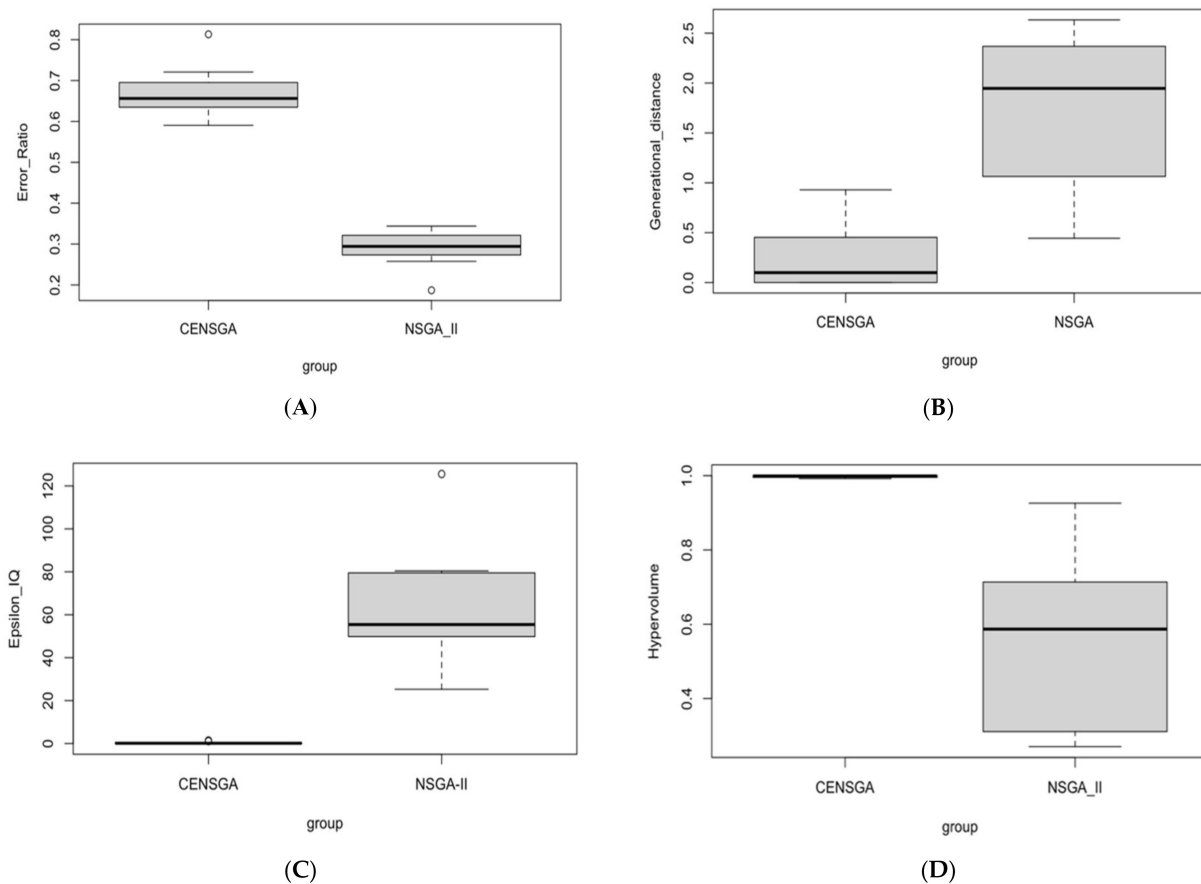
### Appendix A

**Table A1.** Suggested epidemic experiments with their origin sources.

#	Type of Influenza	Location	$R_0$ value	Estimated $R_0$	Estimated Experiments
1	2009 influenza A/H1N1 pandemic	Denmark [58]	2.8–4.0	3.5	$\gamma = 1/5, \beta = 1.18$
2	SARS-CoV	Hong Kong, Vietnam, Singapore, and Canada [59]	3.1–4.5	4.2	$\gamma = 1/5, \beta = 2.36$
3	1918 Influenza A/H1N1 Pandemic	Denmark [58]	3.6–5.4	4.73	$\gamma = 1/5, \beta = 4.72$
4	MERS-COV	Saudi Arabia, South Korea [59]	2.0–7.0	6.88	$\gamma = 1/7, \beta = 1.18$
5	COVID-19 Alpha	Iran [60]	2.26–12.13	8.26	$\gamma = 1/7, \beta = 2.36$
6	COVID-19 Delta	Iran [60]	3.0–23.3	9.29	$\gamma = 1/7, \beta = 4.72$
7	COVID-19 Omicron	South Africa [61]	1.5–24.0	13.77	$\gamma = 1/9, \beta = 1.18$
8	Seasonal influenza epidemics (H1N1)	United Kingdom [58]	16.0–21.0	16.52	$\gamma = 1/9, \beta = 2.36$
9	Seasonal influenza epidemics (H1N1)	United Kingdom [58]	16.0–21.0	18.59	$\gamma = 1/9, \beta = 4.72$



**Figure A1.** Flowchart of the CENSGA algorithm.



**Figure A2.** Performance measures of all mean experiment cases. (A) Error ratio. (B) Generational distance. (C)  $\varepsilon$ -indicator. (D) Hypervolume.

## References

1. Paget, J.; Spreuwenberg, P.; Charu, V.; Taylor, R.J.; Iuliano, A.D.; Bresee, J.; Simonsen, L.; Viboud, C. Global mortality associated with seasonal influenza epidemics: New burden estimates and predictors from the GLaMOR Project. *J. Glob. Health* **2019**, *9*, 020421. [CrossRef] [PubMed]
2. Plotkin, S.A.; Orenstein, W.A.; Offit, P.A. (Eds.) *Plotkin's Vaccines*, 7th ed.; Elsevier: Philadelphia, PA, USA, 2018.
3. Cardoso, R.T.N.; Dusse, A.C.S.; Adam, K. Optimal Vaccination Campaigns Using Stochastic SIR Model and Multiobjective Impulsive Control. *Trends Comput. Appl. Math.* **2021**, *22*, 201–220. [CrossRef]
4. World Health Organization, Global Influenza Programme, University of Edinburgh, and World Health Organization, A Manual for Estimating Disease Burden Associated with Seasonal Influenza. 2015. Available online: [http://apps.who.int/iris/bitstream/10665/178801/1/9789241549301\\_eng.pdf?ua=1](http://apps.who.int/iris/bitstream/10665/178801/1/9789241549301_eng.pdf?ua=1) (accessed on 7 March 2020).
5. Cardoso, R.T.N.; Takahashi, R.H.C. Solving Impulsive Control Problems by Discrete-Time Dynamic Optimization Methods. *Trends Comput. Appl. Math.* **2008**, *9*, 21–30. [CrossRef]
6. Da Cruz, A.R.; Cardoso, R.T.N.; Takahashi, R.H.C. Multiobjective synthesis of robust vaccination policies. *Appl. Soft Comput.* **2017**, *50*, 34–47. [CrossRef]
7. Dusse, A.C.S.; Cardoso, R.T.N. Using a Stochastic SIR Model to Design Optimal Vaccination Campaigns via Multiobjective Optimization. In *Trends in Biomathematics: Modeling Cells, Flows, Epidemics, and the Environment*; Mondaini, R.P., Ed.; Springer International Publishing: Cham, Switzerland, 2020; pp. 245–258. [CrossRef]
8. Hethcote, H.W. The Mathematics of Infectious Diseases. *SIAM Rev.* **2000**, *42*, 599–653. [CrossRef]
9. Hinman, A. Eradication of Vaccine-Preventable Diseases. *Annu. Rev. Public Health* **1999**, *20*, 211–229. [CrossRef]
10. Kim-Farley, R. Global Immunization. *Annu. Rev. Public Health* **1992**, *13*, 223–237. [CrossRef]
11. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*; Addison-Wesley Pub. Co.: Reading, MA, USA, 1989.
12. Deb, K.; Goel, T. Controlled Elitist Non-dominated Sorting Genetic Algorithms for Better Convergence. In *Evolutionary Multi-Criterion Optimization*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 67–81.
13. Jaszkievicz, A. A Comparative Study of Multiple-Objective Metaheuristics on the Bi-Objective Set Covering Problem and the Pareto Memetic Algorithm. *Ann. Oper. Res.* **2004**, *131*, 135–158. [CrossRef]

14. Hajipour, V.; Tavana, M.; Santos-Arteaga, F.J.; Alinezhad, A.; di Caprio, D. An efficient controlled elitism non-dominated sorting genetic algorithm for multi-objective supplier selection under fuzziness. *J. Comput. Des. Eng.* **2020**, *7*, 469–488. [\[CrossRef\]](#)
15. Yazdani, M.; Zandieh, M.; Tavakkoli-Moghaddam, R. Evolutionary algorithms for multi-objective dual-resource constrained flexible job-shop scheduling problem. *Opsearch* **2019**, *56*, 983–1006. [\[CrossRef\]](#)
16. Lara-Molina, F.A.; Rosario, J.M.; Dumur, D. Multi-objective optimization of Stewart-Gough manipulator using global indices. In Proceedings of the 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), Budapest, Hungary, 3–7 July 2011; pp. 79–85. [\[CrossRef\]](#)
17. Ghodrathnama, A.; Tavakkoli-Moghaddam, R.; Kalami-Heris, S.M.; Nagy, G. Solving a novel multi-objective uncapacitated hub location problem by five meta-heuristics. *J. Intell. Fuzzy Syst.* **2015**, *28*, 2457–2469. [\[CrossRef\]](#)
18. Abul'Wafa, A.R. Optimization of economic/emission load dispatch for hybrid generating systems using controlled Elitist NSGA-II. *Electr. Power Syst. Res.* **2013**, *105*, 142–151. [\[CrossRef\]](#)
19. Patel, R.; Longini, I.M.; Halloran, M.E. Finding optimal vaccination strategies for pandemic influenza using genetic algorithms. *J. Theor. Biol.* **2005**, *234*, 201–212. [\[CrossRef\]](#)
20. Hu, X.; Zhang, J. Optimizing Vaccine Distribution for Different Age Groups of Population Using DE Algorithm. In Proceedings of the 2013 Ninth International Conference on Computational Intelligence and Security, Emei Mountain, China, 14–15 December 2013; pp. 21–25. [\[CrossRef\]](#)
21. Hu, X.-M.; Zhang, J.; Chen, H. Optimal Vaccine Distribution Strategy for Different Age Groups of Population: A Differential Evolution Algorithm Approach. *Hindawi* **2014**, *2014*, 7. [\[CrossRef\]](#)
22. Da Cruz, A.R.; Cardoso, R.T.N.; Takahashi, R.H.C. Multi-objective Design with a Stochastic Validation of Vaccination Campaigns. *IFAC Proc. Vol.* **2009**, *42*, 289–294. [\[CrossRef\]](#)
23. Jiang, H.; Yang, X.; Yin, K.; Zhang, S.; Cristoforo, J.A. Multi-path QoS-Aware Web Service Composition using Variable Length Chromosome Genetic Algorithm. *Inf. Technol. J.* **2010**, *10*, 113–119. [\[CrossRef\]](#)
24. Brie, A.H.; Morignot, P. Genetic Planning Using Variable Length Chromosomes. In Proceedings of the Fifteenth International Conference on International Conference on Automated Planning and Scheduling, Monterey, VA, USA, 5–10 June 2005; pp. 320–329.
25. Cruz-Piris, L.; Marsa-Maestre, I.; Lopez-Carmona, M.A. A Variable-Length Chromosome Genetic Algorithm to Solve a Road Traffic Coordination Multipath Problem. *IEEE Access* **2019**, *7*, 111968–111981. [\[CrossRef\]](#)
26. Banzhaf, W. (Ed.) *Genetic Programming: An Introduction on the Automatic Evolution of Computer Programs and Its Applications*; Morgan Kaufmann Publishers: San Francisco, CA, USA; Dpunkt-Verlag: Heidelberg, Germany, 1998.
27. Pal, R.; Mukhopadhyay, S.; Chakraborty, D.; Suganthan, P.N. Very high-resolution satellite image segmentation using variable-length multi-objective genetic clustering for multi-class change detection. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, S1319157821003694. [\[CrossRef\]](#)
28. Zhou, A.; Qu, B.-Y.; Li, H.; Zhao, S.-Z.; Suganthan, P.N.; Zhang, Q. Multiobjective evolutionary algorithms: A survey of the state of the art. *Swarm Evol. Comput.* **2011**, *1*, 32–49. [\[CrossRef\]](#)
29. Gunantara, N. A review of multi-objective optimization: Methods and its applications. *Cogent Eng.* **2018**, *5*, 1502242. [\[CrossRef\]](#)
30. Hu, S.; Wu, X.; Liu, H.; Wang, Y.; Li, R.; Yin, M. Multi-Objective Neighborhood Search Algorithm Based on Decomposition for Multi-Objective Minimum Weighted Vertex Cover Problem. *Sustainability* **2019**, *11*, 3634. [\[CrossRef\]](#)
31. Hagens, A.; İnkaya, A.Ç.; Yildirak, K.; Sancar, M.; van der Schans, J.; Acar Sancar, A.; Ünal, S.; Postma, M.; Yeğenoğlu, S. COVID-19 Vaccination Scenarios: A Cost-Effectiveness Analysis for Turkey. *Vaccines* **2021**, *9*, 399. [\[CrossRef\]](#)
32. Kermack, W.; McKendrick, A. A contribution to the mathematical theory of epidemics. *Proc. R. Soc. Lond. Ser. Contain. Pap. Math. Phys. Character* **1927**, *115*, 700–721. [\[CrossRef\]](#)
33. Aletreby, W.T.; Alharthy, A.M.; Faqih, F.; Mady, A.F.; Ramadan, O.E.; Huwait, B.M.; Alodat, M.A.; Lahmar, A.B.; Mahmood, N.N.; Mumtaz, S.A.; et al. Dynamics of SARS-CoV-2 outbreak in the Kingdom of Saudi Arabia: A predictive model. *Saudi Crit. Care J.* **2020**, *4*, 79. [\[CrossRef\]](#)
34. Van den Driessche, P. Reproduction numbers of infectious disease models. *Infect. Dis. Model.* **2017**, *2*, 288–303. [\[CrossRef\]](#)
35. Shulgin, B. Pulse vaccination strategy in the SIR epidemic model. *Bull. Math. Biol.* **1998**, *60*, 1123–1148. [\[CrossRef\]](#)
36. Hill, A.N.; Longini, I.M. The critical vaccination fraction for heterogeneous epidemic models. *Math. Biosci.* **2003**, *181*, 85–106. [\[CrossRef\]](#)
37. Duijzer, L.E.; van Jaarsveld, W.; Dekker, R. The benefits of combining early aspecific vaccination with later specific vaccination. *Eur. J. Oper. Res.* **2018**, *271*, 606–619. [\[CrossRef\]](#)
38. Yang, T. Impulsive control. *IEEE Trans. Autom. Control* **1999**, *44*, 1081–1083. [\[CrossRef\]](#)
39. Bertsekas, D.P. *Dynamic Programming and Optimal Control, Volume 1*, 2nd ed.; Athena Scientific: Belmont, MA, USA, 2000.
40. Ha, V.-P.; Dao, T.-K.; Pham, N.-Y.; Le, M.-H. A Variable-Length Chromosome Genetic Algorithm for Time-Based Sensor Network Schedule Optimization. *Sensors* **2021**, *21*, 3990. [\[CrossRef\]](#)
41. Kim, I.Y.; de Weck, O.L. Variable chromosome length genetic algorithm for progressive refinement in topology optimization. *Struct. Multidiscip. Optim.* **2005**, *29*, 445–456. [\[CrossRef\]](#)
42. Gobeyn, S.; Goethals, P.L.M. A Variable Length Chromosome Genetic Algorithm Approach to Identify Species Distribution Models Useful for Freshwater Ecosystem Management. In *Environmental Software Systems. Computer Science for Environmental Protection*; Hřebíček, J., Denzer, R., Schimak, G., Pitner, T., Eds.; Springer International Publishing: Cham, Switzerland, 2017; Volume 507, pp. 196–208. [\[CrossRef\]](#)

43. Pawar, S.N.; Bichkar, R.S. Genetic algorithm with variable length chromosomes for network intrusion detection. *Int. J. Autom. Comput.* **2015**, *12*, 337–342. [[CrossRef](#)]
44. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [[CrossRef](#)]
45. Deb, K.; Sindhya, K.; Okabe, T. Self-adaptive simulated binary crossover for real-parameter optimization. In Proceedings of the 9th annual conference on Genetic and evolutionary computation (GECCO'07), London, UK, 7–11 July 2007; p. 1187. [[CrossRef](#)]
46. Deb, K.; Deb, D. Analysing mutation schemes for real-parameter genetic algorithms. *Int. J. Artif. Intell. Soft Comput.* **2014**, *4*, 1. [[CrossRef](#)]
47. Mohapatra, P.; Nayak, A.; Kumar, S.K.; Tiwari, M.K. Multi-objective process planning and scheduling using controlled elitist non-dominated sorting genetic algorithm. *Int. J. Prod. Res.* **2015**, *53*, 1712–1735. [[CrossRef](#)]
48. Szymanski, T.G. Hash table reorganization. *J. Algorithms* **1985**, *6*, 322–335. [[CrossRef](#)]
49. Najafi, A.A.; Niaki, S.T.A.; Shahsavari, M. A parameter-tuned genetic algorithm for the resource investment problem with discounted cash flows and generalized precedence relations. *Comput. Oper. Res.* **2009**, *36*, 2994–3001. [[CrossRef](#)]
50. Tavakkoli-Moghaddam, R.; Vazifeh-Noshafagh, S.; Taleizadeh, A.A.; Hajipour, V.; Mahmoudi, A. Pricing and location decisions in multi-objective facility location problem with  $M/M/m/k$  queuing systems. *Eng. Optim.* **2017**, *49*, 136–160. [[CrossRef](#)]
51. Uray, E.; Carbas, S.; Geem, Z.W.; Kim, S. Parameters Optimization of Taguchi Method Integrated Hybrid Harmony Search Algorithm for Engineering Design Problems. *Mathematics* **2022**, *10*, 327. [[CrossRef](#)]
52. Kv, S.; Baskar, J.; Paul, P. Overview on L9 Taguchi Optimizational Method. *Int. J. Adv. Res. Eng. Technol.* **2019**, *10*, 652–658. [[CrossRef](#)]
53. Montgomery, D.C. *Design and Analysis of Experiments*, 8th ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2013.
54. Li, M.; Yao, X. Quality Evaluation of Solution Sets in Multiobjective Optimisation: A Survey. *ACM Comput. Surv.* **2020**, *52*, 26. [[CrossRef](#)]
55. Santos, T.; Xavier, S. A Convergence indicator for Multi-Objective Optimisation Algorithms. *TEMA* **2018**, *19*, 437–448. [[CrossRef](#)]
56. Audet, C.; Bibeon, J.; Cartier, D.; le Digabel, S.; Salomon, L. Performance indicators in multiobjective optimization. *Eur. J. Oper. Res.* **2021**, *292*, 397–422. [[CrossRef](#)]
57. Higgins, J.J. *An Introduction to Modern Nonparametric Statistics*; Brooks/Cole: Pacific Grove, CA, USA, 2004.
58. Biggerstaff, M.; Cauchemez, S.; Reed, C.; Gambhir, M.; Finelli, L. Estimates of the reproduction number for seasonal, pandemic, and zoonotic influenza: A systematic review of the literature. *BMC Infect. Dis.* **2014**, *14*, 480. [[CrossRef](#)]
59. Xu, H.; Zhang, Y.; Yuan, M.; Ma, L.; Liu, M.; Gan, H.; Liu, W.; Lum, G.G.; Tao, F. Basic Reproduction Number of the 2019 Novel Coronavirus Disease in the Major Endemic Areas of China: A Latent Profile Analysis. *Front. Public Health* **2021**, *9*, 575315. [[CrossRef](#)]
60. Sheikhi, F.; Yousefian, N.; Tehranipoor, P.; Kowsari, Z. Estimation of the basic reproduction number of Alpha and Delta variants of COVID-19 pandemic in Iran. *PLoS ONE* **2022**, *17*, e0265489. [[CrossRef](#)]
61. Liu, Y.; Rocklöv, J. The effective reproductive number of the Omicron variant of SARS-CoV-2 is several times relative to Delta. *J. Travel Med.* **2022**, *29*, taac037. [[CrossRef](#)]