Multi-objective Evolutionary Design of Antibiotic Treatments

Gabriela Ochoa[a], Lee A. Christie[b], Alexander E. Brownlee[a], Andrew Hoyle[a]

[a] *University of Stirling, Stirling, Scotland, UK.*
[b] *Robert Gordon University, Aberdeen, Scotland, UK.*

Abstract

Antibiotic resistance is one of the major challenges we face in modern times. Antibiotic use, especially their overuse, is the single most important driver of antibiotic resistance. Efforts have been made to reduce unnecessary drug prescriptions, but limited work is devoted to optimising dosage regimes when they are prescribed. The design of antibiotic treatments can be formulated as an optimisation problem where candidate solutions are encoded as vectors of dosages per day. The formulation naturally gives rise to competing objectives, as we want to maximise the treatment effectiveness while minimising the total drug use, the treatment duration and the concentration of antibiotic experienced by the patient. This article combines a recent mathematical model of bacterial growth including both susceptible and resistant bacteria, with a multi-objective evolutionary algorithm in order to automatically design successful antibiotic treatments. We consider alternative formulations combining relevant objectives and constraints. Our approach obtains shorter treatments, with improved success rates and smaller amounts of drug than the standard practice of administering daily fixed doses. These new treatments consistently involve a higher initial dose followed by lower tapered doses.

Keywords: antibiotic resistance, antimicrobial resistance, AMR, evolutionary computation,, stochastic model

1. Introduction

Since the introduction of Penicillin in 1942, antibiotics have become one of the most commonly prescribed drugs in human medicine; they are also

widely used in agriculture and aquaculture. Our dependency on these drugs, together with their high usage, has provided selection pressures in the bacterial environment encouraging the evolution of bacterial strains that are resistant to antibiotics. Resistant bacteria are not controlled or killed by antibiotics, instead they are able to survive and even multiply in the presence of antibiotics. Well known examples are Methicillin-Resistant Staphylococcus aureus (MRSA), Clostridium difficile (C-diff) and Enterobacteriaceae. The World Health Organization recently stated that antibiotic resistance is a major threat to public health in every region of the world, and immediate action needs to be taken [1]. Efforts to prevent the emergence of antibiotic resistance include large-scale initiatives within hospitals centring on infection control (e.g. quarantine), restricting antibiotic use and antibiotic cycling. These efforts have had varied level of success [2], however, it is estimated that up to 50% of antibiotics usage is inappropriate, often prescribed when not needed, or administered with incorrect dosing and duration [3].

Antibiotic use is the single most important driver of the selection and enrichment of antibiotic resistance, due to both their unnecessary prescription, and by prescribing larger volumes than required. Traditional antibiotic treatments apply a constant daily dose for a fixed period, i.e., take x mg (milligrams) per day for n days. These regimens may be easier to administer, but there is little evidence that they are the most effective. Indeed, some studies in the medical literature show that shorter treatments than the standard practice can be effective against bacterial infections [4, 5]. Moreover, treatments with an initial higher dose (called loading dose) followed by a lower maintenance dose are beneficial in treating patients in critical care medicine [6]. Similarly, tapered regimens are effective when treating C-diff [7, 8]. However, the generality of these findings has not been assessed, nor are these alternative schedules common practice.

The need to optimise antibiotic treatments becomes critical in ensuring the prolonged effectiveness of these drugs. A question thus emerges: how can we reduce the quantity of prescribed antibiotics without reducing their effectiveness for clearing bacterial infections? Paterson et al. [9] tackled this question with a computational approach using a simple genetic algorithm, allowing them to vary the dose each day. The design of antibiotic treatments was formulated as an optimisation problem where candidate solutions were encoded as vectors of dosages per day. Although formulations using bio-inspired algorithms have been proposed for designing cancer chemotherapies [10–16], very little work in the literature uses evolutionary algorithms to

design tailored antibiotic treatments. A recent study by Cicchese et al. [17] uses a genetic algorithm to design regimens to treat Tuberculosis infections, however, the authors assume that doses are fixed across the treatment, and vary instead the frequency of application of multiple drugs.

In Paterson et al. [9] an underlying mathematical model simulates the progression of a bacterial infection, and the single objective function to be minimised aggregates the bacterial load and antibiotic used across treatment. This approach produced regimens of shorter duration than the standard practice, where a high initial dose is followed by an extended tapering of doses. These new regimes consistently improved the success of eradicating infections, used less antibiotic than traditional regimens, and reduced the time to eradication. Here we adopt the model (and parameters values) in [9] and propose to extend this approach by including additional objectives and constraints, considering a multi-objective formulation and using a well-established evolutionary algorithm to design effective antibiotic regimes. The problem is naturally multi-objective as the goal is to maximise the treatment effectiveness while minimising the total drug use and treatment duration.

The article is organised as follows. Section 2 presents the problem formulation including relevant biomedical background, the mathematical model and the problem objectives. Section 3 illustrates how traditional treatments of fixed daily dosages behave according to our model. Section 4 overviews our computational approach and experimental setting. Results are presented and discussed in Section 5, while Section 6 summarises our main findings and suggestions for future work.

2. Problem Formulation

2.1. Biomedical Background

Antibiotic treatments consist of two key variables: the dose and the duration of treatment. For most antibiotics, the manufacturer identifies a standard treatment regimen which is implemented by medical practitioners when prescribing these antibiotics. These traditional regimens usually consist of a fixed dose administered for a specified duration (generally measured in number of days). Drug efficiency studies are used to determine the dose and duration for these treatment regimens. However, one limitation of this approach is that it only provides information for the regimen being analysed and offers no indication for other potential regimens. As discussed in

the introduction, there is evidence that these standard regimens might be suboptimal.

2.2. Mathematical Model

The mathematical model and parameters used follow the work detailed in [9], which we recap here. The antibiotic dosage regime is given as a vector $\mathbf{x} = (x_0\ x_1, x_2, \ldots, x_{n-1})$, where each entry x_i represents the antibiotic dose taken on day i, with $0 \leq x_i \leq 60$ and $x_i \in \mathbb{Z}$. Dosages x_i are restricted to integers, as in practice, fractional amounts of drugs are not used, instead they are normally measured in grams or milligrams. The antibiotic component of the model follows a deterministic equation:

$$\frac{dC}{dt} = -gC$$

Here, C represents the concentration of antibiotic in the host at time t, which decays exponentially at a rate g. Discrete doses of antibiotic, x_{i-1}, are added to the system at times $t_i = 1, 2, 3, ..., n$, with $C(t)$ updated as $C(t_i) \to C(t_i) + x_{i-1}$. Although t can be arbitrary in terms of units, we think of it here in days.

The bacterial population is divided into two groups: an antibiotic-sensitive population, with density S, and a resistant population with density R. A higher concentration of antibiotic is required to eradicate the resistant population. The populations are modelled with a Markov chain approach, specifically the Gillespie Algorithm [18]. The stochastic model contains the following five possible events:

- p_1 or p_2: the birth of a new sensitive or resistant bacterial cell, increasing the respective population by 1,
- p_3 or p_4: the death of a sensitive or resistant bacterial cell (either naturally or antibiotic induced), reducing the respective population by 1,
- p_5: a resistance gene is passed from a resistant cell to a sensitive cell, making the sensitive cell resistant to antibiotics (a process known as horizontal gene transfer [19].)

The birth process is modelled by a logistic growth function, while the death process by summing a linear function (natural death) and a sigmoidal

function (antibiotic induced death). Finally, gene transfer is modelled by a density dependent mass-action function. The rates describing each of the model's events are given in Table 1, with parameter values given in Table 2.

Table 1: Mathematical terms for each of the rates in the model ($S+R$). The third column displays how each term affects the bacterial population. Terms taken from [9].

p_i	Function	$S, R \to ...$
p_1	$r_S S\left(1-q(S+R)\right)$	$S+1, R$
p_2	$r_R R\left(1-q(S+R)\right)$	$S, R+1$
p_3	$\theta S + a_S S\left(1+b_S(C/\hat{C}_S)^{-k}\right)^{-1}$	$S-1, R$
p_4	$\theta R + a_R R\left(1+b_R(C/\hat{C}_R)^{-k}\right)^{-1}$	$S, R-1$
p_5	βSR	$S-1, R+1$

Table 2: List of parameters and values; taken from [9]. (b_i determined by function, to ensure the bacterial growth rate is zero when $C = \hat{C}_i$)

Parameter	Description	Value
r_S, r_R	Replication Rate	2.7726, 2.2181
q	Competition rate	0.001
β	Rate of gene transfer	0.00001
θ	Natural Death Rate	0.2
g	Degradation rate of antibiotic	0.48
a_S, a_R	Max kill rate at high C	4.6726, 4.1181
b_S, b_R	$b_i = a_i/(r_i - \theta)$	0.8163, 1.0406
$\hat{C}_S, \hat{C}_R$	Min inhibitory concentration (MIC)	16, 32
k	Hill coefficient	4
S_0, R_0	Initial bacterial density	700, 100

At a given point in time, one of the five possible events is chosen at random, based on their probabilities $P(i), i \in 1 \ldots 5$, and the associated effect of that event is applied to the bacterial population. At time t, the probability of each event occurring is:

$$P(i) = \frac{p_i}{\sum_{j=1}^{5} p_j}$$

After an event occurs, the time is updated: $t \to t + \Delta t$, where $\Delta t = -\ln(r)/\rho$, r is a uniformly distributed number in $[0, 1)$ and $\rho = \sum_i p_i$. The probabilities are then recalculated and the process repeated. As the time t passes t_i, the next dose of antibiotic, x_{i-1}, is applied to the system.

Each run of the stochastic model is conducted until a number of days after the treatment is completed or until the total bacteria population is eradicated ($S + R = 0$), whichever is sooner. As the treatment duration will be a maximum of $n = 10$ days for our study, we choose to run the simulations until $t = 15$ days, as by this point the antibiotic in the system will have degraded to negligible levels, and should any bacteria be present, their number will increase indicating that the host was not cured. As this is a stochastic model, each single simulation is either a success (infection eradicated) or failure (infection not eradicated) with some probability. Multiple simulation runs give a statistical estimate of this probability, which we call the failure rate. The higher the number of simulations, the more accurate the estimate of the failure rate but the higher the computational cost. We use 1000 simulations for computing the objective function while running the evolutionary algorithm. However, we use a higher number of simulations 100 000 for re-evaluating the best solutions found by the evolutionary algorithm (Section 4.5) and in our preliminary computations of traditional treatments (Section 3).

2.3. Objectives

Our approach considers three objective functions to be minimised:

1. The percentage of simulation runs where the bacteria survives the treatment, the *failure rate*, f_{fr}.

2. The total amount of antibiotic used, as measured by the sum of the entries in the dosage vector, f_{ta}.

3. The maximum concentration that the antibiotic reaches at any point in time across the simulation, f_{mc}.

The first two objectives, f_{fr} and f_{ta}, are equivalent to those in [9]. The maximum antibiotic concentration was originally a constraint in [9], and was not allowed to exceed 60 mg/l (milligrams/litre); this is due to the toxic nature of antibiotics. However, here we take it as our third objective, f_{mc}, allowing a more thorough search of the solution space and the possibility of

gaining more effective solutions that may have only breached the previous constraint by a small amount.

In addition to the three objectives above, the treatment duration, f_{td} is measured by the number of treatment days before the trailing zeros as follows:

$$\mathbf{x} = [\overbrace{60,\ 30,\ 30,\ 0,\ 0,\ 10}^{f_{\text{td}}(\mathbf{x}) = 6 \text{ days}}, 0, 0, 0, 0]$$

In a clinical setting, longer treatments are correlated with lower patient compliance, therefore, shorter treatments are generally preferable. We model this preference by establishing a maximum length n of the dosage vector $\mathbf{x}$, which represents an upper bound for the treatment duration. Since the treatment duration is measured in days, the range of possible values for this measurement is small. Antibiotic treatments shorter than 4 days never occur in practice; they are not advisable as bacteria need several rounds of treatment to be eradicated. On the other extreme, treatments longer than 10 days are not generally prescribed. In order to widely explore the feasibility of treatments shorter than 10 days. Values of n from 2 to 10 were used in separate runs of our experiments.

The goal of designing an effective antibiotic treatment is formulated as finding a dosage vector

$$\mathbf{x} = \begin{bmatrix} x_0 & x_1 & x_2 & \dots & x_{n-1} \end{bmatrix}$$

which minimises the following objective functions:

$$\begin{aligned} f_{\text{fr}}(\mathbf{x}) &= P\left((S+R)|_{t=15} \neq 0 \right) \\ f_{\text{ta}}(\mathbf{x}) &= \sum_{i=0}^{n-1} x_i \\ f_{\text{mc}}(\mathbf{x}) &= \max_{t \in [0,15]} (C(t)) \ = \ \max_{n=1..10} \left[\sum_{i=1}^{n} x_{i-1} e^{(n-i)g} \right] \end{aligned}$$

Given the exponential decay of antibiotic, the peaks of antibiotic concentration within the host will occur when each dose is applied. Hence the concentration $C(t)$ within this last objective, can be written as a straightforward summation. Although objectives 2 and 3 are linear in terms of x_i,

objective 1 is non-linear, as it is the result of the outcome of a stochastic, non-linear mathematical model.

The detailed implementation of these objectives is described further in Section 4.2 and the choice of objectives in our alternative formulations is described in Section 4.3.

3. Exploring Traditional Treatments

In order to provide a benchmark for our study, this section explores the behaviour of constant daily dosage treatments using the model. We enumerated all treatments with a constant daily dose of up to 240 mg of duration 2 to 10 days. More formally, $\mathbf{x} = (x_0 \ldots, x_{n-1})$ with $x_0 = x_1 = ... = x_{n-1}$ where $2 \leq n \leq 10$ and $1 \leq x_i \leq 240$ with $x_i \in \mathbb{Z}$. For each treatment, the model considered 100 000 simulation runs to compute the failure rate objective. We report only successful treatments, that is treatments where the failure rate is lower than 1%; this was chosen to make the treatments more viable in clinical settings. The results are given in Table 3, where red font highlights treatments where the maximum concentration exceeds 60 mg/l (which was a constraint in [9]). As the results in Table 3 indicate, to reach a successful treatment (failure rate $< 1\%$) without exceeding a maximum antibiotic concentration of 60 mg/l, at least 9 daily doses of 23 mg are required. As the treatment duration gets shorter, the daily dose increases more rapidly, as does the maximum concentration.

The main goal of our formulation using evolutionary algorithms described in the next Section is to find successful treatments improving on the fixed dose practice by allowing daily dosages to vary across the treatment duration.

4. Methods

We consider two and three objective formulations of the antibiotic optimisation problem. Multi-objective evolutionary algorithms are applied over the space of possible treatment schedules to approximate the Pareto-optimal trade-offs.

4.1. Problem Encoding

The dosage schedule taken from the mathematical model, is encoded as an `IntegerSolution` in the Jmetal [20] framework:

$$\mathbf{x} = [x_0, x_1, \ldots, x_{n-1}]$$

Table 3: Minimum fixed daily dosage required for a successful treatment (failure rate lower than $< 1\%$). Red indicates solutions violating the maximum concentration constraint (60 mg), previously imposed by [9].

Duration (days)	Fixed Dosage (mg)	Max. Concentration (mg/l)	Total Antibiotic (mg)
2	91	147.3	182
3	56	112.1	168
4	42	94.0	168
5	35	83.5	175
6	30	74.3	180
7	27	68.4	189
8	25	64.2	200
9	23	59.5	207
10	22	57.2	220

subject to the hard variable range bounds

$$x_i \in \{0, 1, \ldots, 60\}\,.$$

A constraint forcing x_0 to be non-zero was added to ensure the treatments started on day 1, in line with medical expectations. Following [9], most of our experiments set the upper bound for treatment duration of $n = 10$, as 10 days represents the maximum duration of an antibiotic regimen in common medical practice.

4.2. Implementation of the Objectives

The three objectives considered in our formulations are those discussed in Section 2.3, namely, Failure Rate, f_{fr}, Total Antibiotic, f_{ta}, and Maximum Concentration, f_{mc}. We also measure treatment duration.

The failure rate is estimated by running the stochastic model with a fixed number of 1000 simulations, returning the percentage of runs in which the bacteria were not eliminated. The total antibiotic is the sum of the dosages in the candidate solution vector. The maximum concentration is calculated by the deterministic model of antibiotic decay, returning the maximum value achieved across the simulation. The treatment duration is determined from the location of the last non-zero entry in the vector.

Let us consider an example antibiotic treatment vector

$$\mathbf{x} = [60, 30, 30, 30, 0, 0, 0, 0, 0, 0]$$

the failure rate is estimated from the stochastic model, in this case $f_{\text{fr}}(\mathbf{x}) \approx 1.15\%$. The total antibiotic is the sum of the vector components, giving $f_{\text{ta}}(\mathbf{x}) = 150$. The maximum concentration is calculated from modelling the decay of antibiotic. The concentration at each time step is computed and the maximum value is retrieved, in this case $f_{\text{mc}}(\mathbf{x}) = 74.3$. The treatment duration is 4, since there are four treatment days before the vector terminates in trailing zeros.

The required simulations for each evaluation take around 2–3 seconds to complete on an i7-3820 CPU @ 3.60GHz. This motivates the use of an evolutionary algorithm: the search space is 61^{10}, and even greatly reducing it by having doses in increments of 10, the space is 7^{10} which would require 2.8×10^8 evaluations, or approximately 9800 days of CPU time.

4.3. Alternative Multi-objective Formulations

A successful treatment should have a low failure rate, therefore, minimising the failure rate was kept as an objective in all our formulations. We initially considered two formulations with two objectives, namely, pairing Failure Rate with both Maximum Concentration ($f_{\text{fr}}, f_{\text{mc}}$), and Total Antibiotic ($f_{\text{fr}}, f_{\text{ta}}$). However, our main formulation incorporates the three objectives, namely, Failure rate, Maximum concentration and Total Antibiotic ($f_{\text{fr}}, f_{\text{mc}}, f_{\text{ta}}$). For each formulation two sets of experiments were conducted, with and without setting a constraint on the failure rate to be lower than 1%. Our expectation is that this stringent constraint on the failure rate will help the algorithm to focus the search on successful treatments.

The formulations with two objectives consider an upper bound of treatments duration of $n = 10$. However, for the formulation involving the complete set of three objectives, separate experiments considered $n \in \{2, 3, \ldots, 10\}$ in order to restrict the search space and thus allow the algorithm to explore more thoroughly shorter treatment schedules.

4.4. Evolutionary Algorithm

Multi-objective optimisation has vast practical importance because most real-world optimisation problems are naturally posed as non-linear programming problems having multiple conflicting objectives. Such problems give rise

to a set of trade-off optimal solutions (known as *Pareto-optimal* solutions, or Pareto front) instead of a single optimum solution. It becomes important to find as many Pareto optimal solutions as possible, as each of them represents a different trade-off between objectives. Users will be in a better position to select the most suitable solution when many trade-off solutions are revealed. Population-based approaches (evolutionary algorithms) have proven to be an efficient way of finding multiple Pareto-optimal solutions simultaneously in a single algorithm run. Many evolutionary approaches have been proposed, making research and applications of multi-objective evolutionary algorithms quite active and successful over the last two decades. The interested reader may find numerous introductory tutorials [21, 22] and survey articles [23–25].

We used the Non-dominated Sorting Genetic Algorithm II (NSGA-II), one of the best-known and frequently-used multi-objective evolutionary algorithms [26]. NSGA-II uses non-dominated sorting of individuals in the population, with a crowding distance penalty applied to individuals to maintain a diverse Pareto front. The optimal front found is simply the set of non-dominated individuals in the final population. The focus of the present paper is the application rather than the algorithm, so NSGA-II was chosen simply as it is a well-established approach for 2 and 3 objective problems, with proven success shown in numerous surveys of MOEA applications (e.g. [23]). We made use of the Jmetal 5 suite [20] for its NSGA-II implementation and experimental framework.

The NSGA-II parameters were set to typical values given by Tables 4 and 5. These were chosen to yield a reasonable trade-off between solution quality and execution time. As the goal of this paper is to propose a novel application domain, rather than an investigation of algorithm performance, no further parameter tuning was conducted. Each experiment was run 30 times, and Pareto fronts were generated by cumulating the non-dominated solutions across the 30 runs.

Table 4: List of algorithm operators.

Description	Value
Selection	Binary Tournament Selection
Comparator	Ranking and Crowding Distance Comparator
Mutation	Integer Polynomial Mutation
Crossover	Integer SBX Crossover

Table 5: List of algorithm parameters and values.

Parameter	Value
Population Size (n)	100
Mutation Probability	$1/n$
Mutation Distribution Index	20
Crossover Probability	0.9

4.5. Re-evaluation of Best-found Solutions

During the evolutionary algorithm runs, the failure rate determined by the stochastic model was susceptible to noise due to two factors. First, the selected number of 1000 simulations per fitness evaluation, necessary to reduce the computational effort. Second, an elitist bias intrinsic to the the NSGA-II algorithm as it selects solutions after evaluating them once, then reselects (due to elitism) those candidates for which noise could have produced an optimist estimate. This can mean that some of the final solutions obtained are not Pareto-optimal. However, this effect can be mitigated by re-evaluating the failure rate after the completion of the evolutionary run, using a larger number of simulations (100 000) per fitness evaluation, and recomputing the Pareto front. Thus, during the evolutionary process, the lower number of simulations effectively forms a "Problem Approximation" surrogate with "No Evolution Control" [27, 28] for the true fitness function (the larger number of simulations). The resulting re-evaluated failure rates were used in the plots and for reporting the results as this produces a more accurate assessment of the solutions obtained.

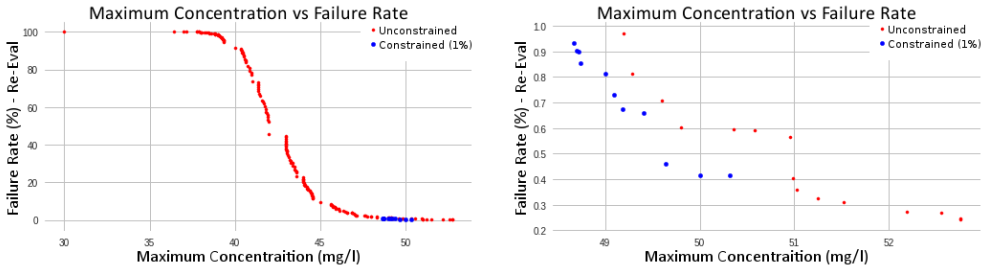
It must also be noted that the final results have undergone a cleaning process, whereby any doses lower than 5 mg were flattened to zero before the objectives were re-evaluated. This reduces the stochastic noise inherent to results from evolutionary search. Moreover, this makes little difference from the application perspective, as it is understood that such small doses have negligible impact on both host and infection.

5. Results

Results are grouped by problem formulation, specifically which objectives and constraints were considered. For each configuration, we report the approximations to the Pareto-optimal front and discuss the trends observed.

Recall that Pareto fronts were generated by cumulating the non-dominated solutions across 30 runs for each experiment.

5.1. Two Objectives (f_{fr}, f_{mc})



(a) all solutions (b) zoom in solutions with failure rates $< 1\%$

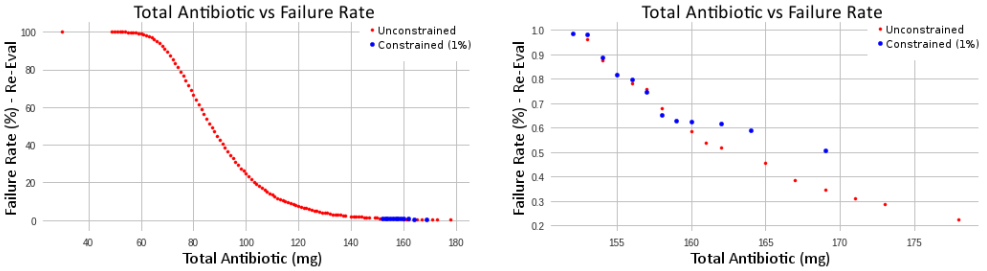
Figure 1: Failure Rate vs Maximum Concentration formulation. Pareto front across 30 runs.

First we give the results for Failure Rate vs Maximum Concentration. These runs consider an upper bound for the treatment duration of $n = 10$. Figure 1 gives the Pareto front plotted in objective space. The red points are the approximated Pareto front from the unconstrained runs, and the blue points are those from the constrained runs (i.e with $f_{\text{fr}}(\mathbf{x}) < 1\%$). The right plot is a zoomed part of the front around the solutions with Failure Rate lower than 1%. Plot (b) illustrates that the solutions obtained with the constrained runs (blue dots) can dominate those obtained from the unconstrained runs (red dots), with the improvement sometimes being quite large. Interestingly, the unconstrained approach is able to find solutions with lower failure rates than the constrained one, those solutions with failure rates of less than 0.4% here. As can be seen in Figure 1 (b), for successful solutions (failure rate below 1%), there is not much of a trade-off with maximum concentration, which needs to increase considerably to achieve a further reduction in failure rate.

One further observation (Figure 1 (a)) is that there is a notable discontinuity in the front around a maximum concentration $f_{\text{mc}}(\mathbf{x}) = 42$. The same effect is apparent to a lesser extent around $f_{\text{mc}}(\mathbf{x}) = 39$ and $f_{\text{mc}}(\mathbf{x}) = 45$. Further analysis revealed that this is down to the integer encoding in combination with the nature of the maximum concentration objective. There is no Pareto optimal way to obtain a value in these regions: increasing the first

dosage in treatment by 1 mg brings a steep increase in the maximum concentration and a corresponding decrease in the failure rate, and decreasing later dosages has no effect on the maximum concentration (if they are reduced, the antibiotic concentration will drop off, but the maximum has already been set).

5.2. Two Objectives (f_{fr}, f_{ta})



(a) all solutions (b) zoom in solutions with failure rates $< 1\%$

Figure 2: Failure Rate vs Total Antibiotic formulation. Pareto front across 30 runs.

Figure 2 shows the Failure Rate vs Total Antibiotic Pareto front. As in the previous experiment 30 runs are conducted with an upper bound for treatment duration of $n = 10$. The right plot shows the same front zoomed into the solutions with Failure Rate lower than 1%. As before, the red points are the approximated Pareto front from the unconstrained runs, and the blue points are those from the constrained runs (i.e with $f_{\text{fr}}(\mathbf{x}) < 1\%$). This Pareto front (plot (a)) is much smoother when compared with the Failure Rate vs. Maximum Concentration front (Fig. 1 (a)), and there is no visible discontinuity. This is because the impact of the integer encoding is lesser here; increments in the antibiotic amount at any given day affect the Total Antibiotic objective. Figure 2 (b) indicates that not always the solutions obtained from the constrained runs (blue dots) dominate those obtained by the unconstrained runs (red dots). Similarly than in the previous section, the unconstrained approach finds solutions with lower failure rates by having higher total antibiotic.

5.3. Three Objectives, Unconstrained

For the formulation considering the three objectives, separate runs were conducted for the upper bound on treatment durations n varying from 2 to 10.

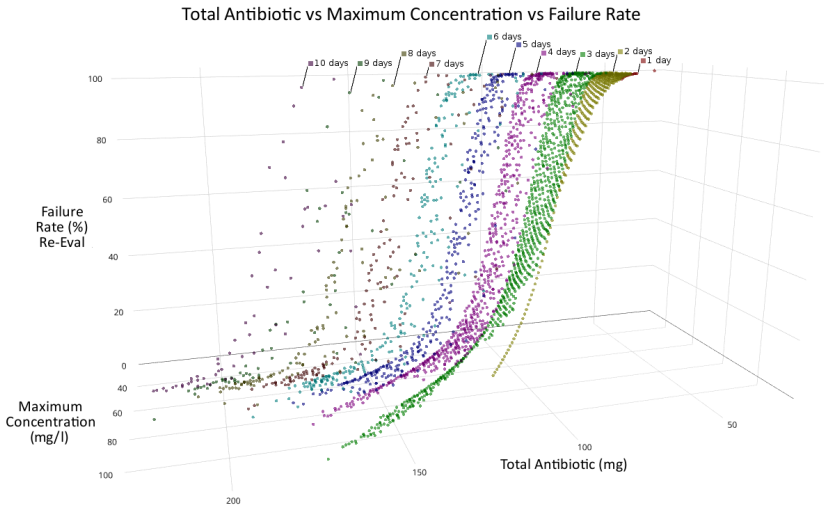
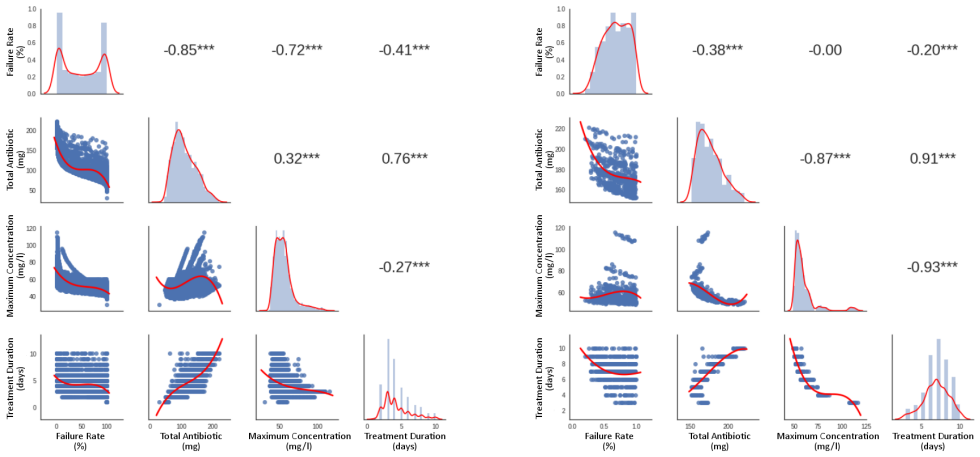


Figure 3: Pareto front for three-objective optimiser over 30 runs. Points representing treatments are coloured by actual treatment duration.

The idea is to thoroughly explore the space of possible shorter treatments. The combined Pareto front from the unconstrained approach is shown in Figure 3. Each point represents a treatment, with the colour indicating the treatment duration. There is a wide spread of treatment strategies, with clear bands in the treatment duration. Longer treatments allow a lower maximum concentration for the same total antibiotic, while maintaining the same failure rate. The plot also shows that failure rate only drops below our target of 1% with treatments of 3 days or longer.

Figure 4 (a) shows the Spearman correlation matrix between the three objectives and the treatment duration for the solutions in the Pareto front. Within the figure, the lower-left triangle shows 2D scatter plots of the treatments for each pair of objective; the diagonal, histograms with the distribution of values for each single objective; and the top-right triangle the correlation coefficients with three stars indicating statistical significance ($p < 0.001$). There is a strong positive correlation between treatment duration and total antibiotic, which is to be expected as longer treatments require larger amounts of drugs. The failure rate is strongly negatively correlated with both the total antibiotic and the maximum concentration. However, the negative correlation between the failure rate and the treatment duration is not strong, which suggests that effective treatments with shorter duration can



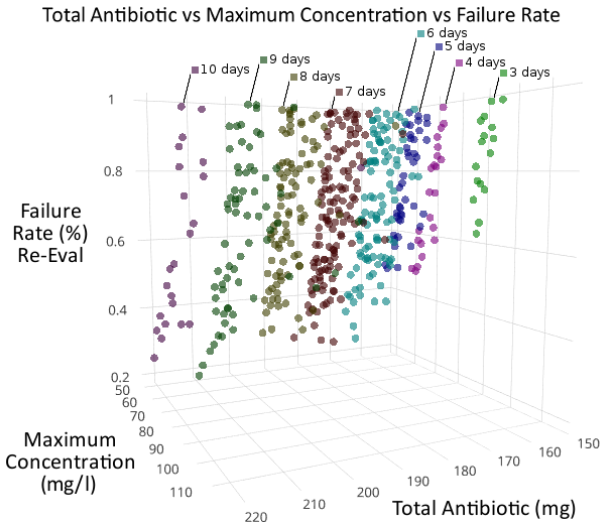
(a) unconstrained (b) constrained (failure rates $< 1\%$)

Figure 4: Pareto front pair-wise correlations between objectives and treatment duration for the three-objective formulation. The lower-left triangle shows 2D scatter plots, the diagonal histograms with the distribution of values for each objective, and the top-right triangle the correlation coefficients. Variables are (from left-to-right, or top-to-bottom): Failure Rate (%), Total Antibiotic (mg), Maximum Concentration (mg/l), Treatment Duration (days).

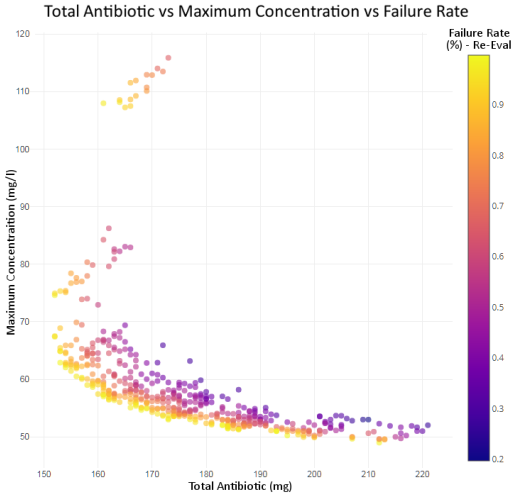
be obtained. This motivates the final experiment, where the run is repeated while applying a constraint on the failure rate to explore only solutions with a value below 1%.

5.4. Three Objectives, Constrained

Figure 5 (a) illustrates the Pareto front for the constrained formulation (failure rate lower than 1%). Each point represents a treatment, with the colour indicating the treatment duration. Again, clear bands for each treatment duration can be observed, revealing a linear relationship between total antibiotic and duration. The correlation matrix (Fig. 4 (b)) captures the trends between the objectives within the Pareto front more clearly. The positive correlation between duration and total antibiotic is stronger as compared to the unconstrained formulation. However, the strong negative correlations observed on the unconstrained formulation (Fig. 4 (a)) between the failure rate with both the total antibiotic and the maximum concentration, are no longer present in the constrained case (Fig. 4 (b)). This is encouraging as it suggests that in the region of successful treatments, several alternatives can be found which do not compromise the treatment success.



(a) 3D view; coloured by treatment duration.



(b) 2D view; coloured by failure rate.

Figure 5: Pareto front for the three-objective constrained formulation, with treatment durations of 3 to 10.

Fig. 4 (b) also shows that the total antibiotic is strongly positively correlated with the treatment duration; in contrast, the maximum concentration is strongly negatively correlated with the duration. This means that the range of treatments span from having high doses in a short time, to having lower

doses spread out over a longer time. This trade-off is clearer in Figure 5 (b), where we re-orient the Pareto front plot to a top-down version of Figure 5, showing only the trade-off between Total Antibiotic and Maximum Concentration. It is clear that, for treatments with low maximum concentration, as the concentration increases, the total antibiotic and treatment length drop, indicating the treatments shift to a higher-intensity but shorter schedule. This trade-off is smooth without any obvious knee-points or discontinuities.

Table 6: Selection of dosage schedules produced by constrained 3-objective optimisation.

	Dosage Vector (mg)										Tot. Antib. (mg)	Max. Conc. (mg/l)	Fail. Rate (%)
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9			
V0	60	34	27	31							152	74.941	0.94
V1	59	25	25	23	23						155	63.060	0.86
V2	58	31	23	27	22						161	66.889	0.57
V3	58	22	20	24	17	20					161	58.541	0.89
V4	57	21	22	21	22	20					163	57.000	0.83
V5	52	30	22	22	22	19					167	62.177	0.53
V6	54	21	20	20	21	21	20				177	54.414	0.62
V7	53	20	20	19	21	21	20	19			193	53.748	0.31
V8	51	20	19	21	19	21	19	21	19		210	52.996	0.20
V9	47	20	20	20	20	18	20	18	18	18	219	51.663	0.32

Table 7: Table of fixed dosage schedules - for comparison with the optimised dosage schedules.

	Dosage Vector (mg)										Tot. Antib. (mg)	Max. Conc. (mg/l)	Fail. Rate (%)
	x_0	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9			
F0	42	42	42	42							168	94.021	0.98
F1	35	35	35	35	35						175	83.482	0.85
F2	30	30	30	30	30	30					180	74.278	0.96
F3	27	27	27	27	27	27	27				189	68.366	0.87
F4	25	25	25	25	25	25	25	25			200	64.170	0.73
F5	23	23	23	23	23	23	23	23	23		207	59.531	0.99
F6	22	22	22	22	22	22	22	22	22	22	220	57.235	0.80

Ultimately, if these treatments are to be applied in practice, we must consider what they look like in terms of the actual doses administered. Table 6 shows the details of a selection of effective treatments taken from the constrained 3-objective Pareto front, sorted in ascending order of the Total

Antibiotic objective. We analysed the complete Pareto front and selected a set of 10 effective solutions with durations ranging from 4 to 10 days, as a representative sample reflecting the trends observed. For comparison purposes, we also report in Table 7, the best treatments of the same duration that can be obtained with fixed daily doses, as explored in Section 3.

Several observations can be made on the treatments. Across all the treatments in Table 6 (and indeed all the Pareto front), the first dose is consistently high compared to the rest of the treatment. This is by a substantial margin: usually 2–2.5× the level of the second dose. This corroborates the earlier finding in [9].

Moving down the set of treatments in Table 6 in increasing order of $f_{\mathrm{ta}}(\mathbf{x})$, each time the treatment duration increases by one day (e.g. V2 to V3), the treatments show a "cliff-edge", with a high first dose (V3 starts with 58 mg) then flat doses for the rest of the treatment (around 20 mg for V3). Treatments then gradually shift to a tapered regime (e.g. V5), with a smoother drop from the high first dose (V5 has 52 mg, then 30 mg, then constant around 20 mg). The pattern then repeats, with an increase in duration and a return to the cliff-edge (e.g. V6, which has a first does of 54 mg followed by all doses around 20 mg). This tapering effect is less clearly visible for the longer treatments.

Some explanation for this pattern lies in the higher concentrations more effectively tackling the infection. The high initial dose pushes the host to a high concentration level, then the rest of the schedule tops up the antibiotic load by the same amount every day to keep the concentration high, without exceeding the maximum.

For schedules of the same duration (e.g. V3 and F2), once the doses have moved to a constant level after the first dose, the doses are always lower with the optimised schedules than for the constant schedules.

Comparing the objective values between the optimised schedules in Table 6 and fixed dosage schedule in Table 7, we find that the optimised schedules achieve better values across the objectives than the fixed schedules. Solutions V0–V5 all have total antibiotic levels below the minimum found when using the constant dose scheme (F0, with 168 mg), yet their failure rates are still lower than that for F0. Any constant dose scheme with total antibiotic levels below 168 mg have a failure rate in excess of 1%. The same is true of the solutions with the lowest maximum concentration: V6–V9 have maximum concentrations of 54.414 mg/l or less; yet the lowest maximum concentration for the fixed dosage schedules is 57.235 mg/l.

For schedules with a comparable failure rate, both the other objectives are matched or improved with the tapered schedules. e.g. V0 in Table 6 has a failure rate of 0.94%; the closest in Table 7 is F2 with a failure rate of 0.96%. F2 has a similar maximum concentration to V0, but higher total antibiotic of 180 mg compared to 152 mg. V3 and F3 have failure rates of 0.89% and 0.87%; yet V3 has lower total antibiotic (161 mg vs 189 mg) and maximum concentration (58 mg vs 68 mg).

Overall, the optimised treatment schedules surpass the fixed dosage schedules in each objective, with only the small change of increasing the first one or two doses and decreasing the rest.

6. Conclusion

As far as we are aware, this study is the first to use multi-objective evolutionary algorithms and constrained optimisation for the automatic design of effective antibiotic treatments. Previous studies using evolutionary algorithms for optimising antibiotic treatments [9, 17] considered standard genetic algorithms and appeared in general science or biomedical journals. Therefore, the present article brings this relevant problem to the attention of the computational intelligence community. We have explored formulations with two and three objectives, modelling the most important factors to design successful antibiotic treatments. Specifically, the goal is to firstly minimise the failure rate, but to do so while also minimising the maximum concentration of the antibiotic at any given point, and the total antibiotic used. We also measured the treatment duration in days, in order to find treatments which are also short, and thus of less burden to the patient. Reducing the total antibiotic used on any given treatment is relevant to controlling the emergence of resistant bacterial strands. Our results suggest that evolutionary algorithms can be used to design successful treatments by widely exploring the large space of alternative solutions and trade offs. Optimising the daily dosages produced shorter treatments, with improved success rates and smaller amounts of drug than the standard practice of administering daily fixed doses. The evolved treatments consistently administered a higher dose on the first day or two, followed by lower maintenance dosages.

As an inter-disciplinary project, our future work can follow three main lines of research. First, it is important to validate our findings in the real-world, indeed experiments are currently on the way for testing the effectiveness of the evolved treatments in the (biology) lab. Secondly, the mathe-

matical model can be extended to include additional relevant aspects such as multiple drugs, different levels of bacterial resistance, and more realistic antibiotic decay. Finally, from the computational perspective, we can apply methods to handle noisy fitness functions, surrogate models and additional multi-objective evolutionary algorithms.

References

[1] W. H. Organization, et al., Global action plan on antimicrobial resistance. 2015, 2017.

[2] van Duijn et. al., The effects of antibiotic cycling and mixing on antibiotic resistance in intensive care units: a cluster-randomised crossover trial, The Lancet Infectious Diseases 18 (2018) 401–409.

[3] C. L. Ventola, The Antibiotic Resistance Crisis: Part 1: Causes and Threats, Pharmacy and Therapeutics 40 (2015) 277–283.

[4] S. Briggs, R. Ellis-Pegler, S. Roberts, M. Thomas, A. Woodhouse, Short course intravenous benzylpenicillin treatment of adults with meningococcal disease, Intern. Med. J. 34 (2004) 383387.

[5] C. M. e. a. Parry, Randomized controlled comparison of ofloxacin, azithromycin, and an ofloxacin-azithromycin combination for treatment of multidrug-resistant and nalidixic acid-resistant typhoid fever, Antimicrob. Agents Chemother. 51 (2007) 819825.

[6] C. McKenzie, Antibiotic dosing in critical illness, Journal of Antimicrobial Chemotherapy 66 (2011) ii25–ii31.

[7] L. V. McFarland, G. W. Elmer, C. M. Surawicz, Breaking the cycle: treatment strategies for 163 cases of recurrent Clostridium difficile disease, American Journal Of Gastroenterology 97 (2002) 1769.

[8] L. V. McFarland, Alternative treatments for clostridium difficile disease: What really works?, J. Med. Microbiol. 54 (2005) 101111.

[9] I. K. Paterson, A. Hoyle, G. Ochoa, C. Baker-Austin, N. G. H. Taylor, Optimising antibiotic usage to treat bacterial infections, Scientific Reports 6 (2016) 37853 EP –.

[10] M. Villasana, G. Ochoa, Heuristic design of cancer chemotherapies, IEEE transactions on Evolutionary Computation 8 (2004) 513–521.

[11] G. Ochoa, M. Villasana, E. Burke, An evolutionary approach to cancer chemotherapy, Genetic Programming and Evolvable Machines 8 (2007) 301–318.

[12] M. Villasana, G. Ochoa, S. Aguilar, Modeling and optimization of combined cytostatic and cytotoxic cancer chemotherapy, Artificial Intelligence in Medicine 50 (2010) 163–173.

[13] A. Petrovski, B. Sudha, J. McCall, Optimising cancer chemotherapy using particle swarm optimisation and genetic algorithms, in: Parallel Problem Solving from Nature - PPSN VIII, volume 3242 of *LNCS*, Springer, Berlin, 2004, pp. 633–641.

[14] A. Petrovski, S. Shakya, J. McCall, Optimising cancer chemotherapy using an estimation of distribution algorithm and genetic algorithms, in: M. Cattolico (Ed.), Genetic and Evolutionary Computation Conference, GECCO 2006, volume 1, ACM Press, New York, 2006, pp. 413–418.

[15] S.-M. Tse, Y. Liang, K.-S. Leung, K.-H. Lee, T.-S.-K. Mok, A memetic algorithm for multiple drugs cancer chemotherapy schedule optimization, IEEE Transcations on Systems, Man, and Cybernetics-Part B: Cybernetics 37 (2007) 84–91.

[16] A. Petrovski, A. Brownlee, J. McCall, Statistical optimisation and tuning of GA factors, in: Proc. IEEE CEC, volume 1, IEEE Press, 2005, pp. 758–764.

[17] J. M. Cicchese, E. Pienaar, D. E. Kirschner, J. J. Linderman, Applying optimization algorithms to tuberculosis antibiotic treatment regimens, Cellular and Molecular Bioengineering 10 (2017) 523–535.

[18] D. T. Gillespie, A general method for numerically simulating the stochastic time evolution of coupled chemical reactions, Journal of Computational Physics 22 (1976) 403 – 434.

[19] S. Sørensen, M. Bailey, L. Hansen, N. Kroer, S. Wuertz, Studying plasmid horizontal transfer in situ: a critical review, Nature Reviews Microbiology 3 (2005) 700–710.

[20] A. J. Nebro, J. J. Durillo, M. Vergne, Redesigning the jmetal multi-objective optimization framework, in: Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation, GECCO Companion '15, ACM, New York, NY, USA, 2015, pp. 1093–1100.

[21] K. Deb, Multi-objective optimization, in: E. K. Burke, G. Kendall (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, Springer US, Boston, MA, 2014, pp. 403–449.

[22] M. T. M. Emmerich, A. H. Deutz, A tutorial on multiobjective optimization: fundamentals and evolutionary methods, Natural Computing 17 (2018) 585–609.

[23] B. Qu, Y. Zhu, Y. Jiao, M. Wu, P. N. Suganthan, J. Liang, A survey on multi-objective evolutionary algorithms for the solution of the environmental/economic dispatch problems, Swarm and Evolutionary Computation 38 (2018) 1–11.

[24] M. Reyes-Sierra, C. Coello, Multi-objective particle swarm optimizers: A survey of the state-of-the-art, International Journal of Computational Intelligence Research 2 (2006) 287–308.

[25] A. Trivedi, D. Srinivasan, K. Sanyal, A. Ghosh, A survey of multiobjective evolutionary algorithms based on decomposition, IEEE Transactions on Evolutionary Computation 21 (2017) 440–462.

[26] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE T. Evolut. Comput. 6 (2002) 182–197.

[27] Y. Jin, A comprehensive survey of fitness approximation in evolutionary computation, Soft Comput. 9 (2005) 3–12.

[28] Y. Jin, Surrogate-assisted evolutionary computation: Recent advances and future challenges, Swarm and Evolutionary Computation 1 (2011) 61 – 70.