Artificial Intelligence In Medicine 133 (2022) 102405

Contents lists available at ScienceDirect

Artificial Intelligence In Medicine

journal homepage: www.elsevier.com/locate/artmed

Research paper

Evolutionary optimisation of antibiotic dosing regimens for bacteria with different levels of resistance

Mila Goranova *, Gabriela Ochoa, Patrick Maier, Andrew Hoyle

Computing Science and Mathematics, University of Stirling, Scotland, UK

ARTICLE INFO

Keywords:
Antimicrobial resistance
Evolutionary algorithms
Differential evolution
Optimisation
Mathematical modelling
Pharmacokinetics/pharmacodynamics modelling
MIC
Antibiotic dosing regimens

ABSTRACT

Antimicrobial resistance is one of the biggest threats to global health, food security, and development. Antibiotic overuse and misuse are the main drivers for the emergence of resistance. It is crucial to optimise the use of existing antibiotics in order to improve medical outcomes, decrease toxicity and reduce the emergence of resistance. We formulate the design of antibiotic dosing regimens as an optimisation problem, and use an evolutionary algorithm suited to continuous optimisation (differential evolution) to solve it. Regimens are represented as vectors of real numbers encoding daily doses, which can vary across the treatment duration. A stochastic mathematical model of bacterial infections with tuneable resistance levels is used to evaluate the effectiveness of evolved regimens. The objective is to minimise the treatment failure rate, subject to a constraint on the maximum total antibiotic used. We consider simulations with different levels of bacterial resistance, two ways of administering the drug (orally and intravenously), as well as coinfections with two strains of bacteria. Our approach produced effective dosing regimens, with an average improvement in lowering the failure rate 30%, when compared with standard fixed-daily-dose regimens with the same total amount of antibiotic.

1. Introduction

Antibiotics are one of the most commonly prescribed drugs not only in human health but also in animal health and agriculture [1,2]. Many decades after the discovery of the first antibiotics, bacterial infections have again become a global threat [3]. The World Health Organisation (WHO) has stated that "antimicrobial resistance is a global crisis that threatens a century of progress in health and achievement" [4]. A recent study using machine learning to analyse antimicrobial resistance research trends reveals that the number of publications in this topic increased by 450% between 1999 and 2018, a testimony of its growing relevance [5]. The overuse and misuse of antibiotics are driving the evolution of resistant bacteria strains. To deal with this complex threat, a range of approaches are required, including not only the development of new antibiotics, but critically novel strategies to optimise the use of existing drugs [6,7]. There are clear opportunities for mathematical modelling and artificial intelligence to contribute to this challenge.

Traditional antibiotic regimes apply a constant daily dose for a fixed number of days. However, medical and biological evidence suggests that regimes with varying daily doses can be more effective. Examples are treatments with an initial higher dose followed by a lower maintenance dose, as well as tapered regimes [8–10]. It is crucial to improve the precision of current antibiotics use. Precision use involves better choice of drugs, but also better dosing and treatment duration. Improving the precision can lead to better clinical outcomes of infectious diseases, while minimising toxicity and the emergence of drug resistance [7].

Evolutionary algorithms have been applied to optimise antibiotic dosing regimens [11–15]. To apply evolutionary algorithms in this context, two key components are required: (i) a representation or encoding of candidate solutions (dosing regimens); and (ii) a fitness (objective) function that measures the quality of the evolved dosing regimens. In order to measure the effectiveness of a regimen, a simulation model of bacterial infection and the effects of drug concentration is required. A problem formulation can also include constraints and/or multiple objectives.

This work uses the stochastic mathematical model of a generic bacterial infection and the effect of an antibiotic agent first introduced in [11]. This initial formulation [11] considered a single aggregated objective to be minimised with terms for the total antibiotic and the proportion of unsuccessful model runs (failure rate). Solutions were encoded as vectors of integer numbers indicating daily dosages, and a simple genetic algorithm was used to optimise dosing regimens. A follow-up work [13] used the same mathematical model and solution encoding, but considered formulations with two and three objectives to be minimised, including the failure rate, the total antibiotic used,

* Corresponding author.
E-mail address: m.g.goranova@stir.ac.uk (M. Goranova).

https://doi.org/10.1016/j.artmed.2022.102405
Received 9 December 2021; Received in revised form 29 July 2022; Accepted 15 September 2022
Available online 24 September 2022

and the maximum antibiotic concentration at any given point during treatment. A classic multi-objective evolutionary algorithm, NSGA-II [16] was used to explore the space of possible dosing regimes to approximate Pareto-optimal trade-offs. A subsequent extension [15], considered a similar multi-objective formulation (with two objectives: failure rate and total antibiotic to be minimised), but added a pharmacodynamics component to the bacterial model to account for the delay of ingested drug to reach the blood stream. Two population based multi-objective optimisation algorithms were contrasted, with the classic NSGA-II producing best results.

Here, we use our most recent bacterial infection model [15], but make two fundamental changes to the optimisation problem formulation. Firstly, the total amount of antibiotic is treated as a constraint rather than an objective. The single objective to be minimised is the regimen failure rate. This is because reducing the failure rate is the prominent aim of any successful treatment, while reducing the total amount of antibiotic used is a secondary goal. Therefore, exploring the whole trade-off of these two goals, as it is done by multi-objective evolutionary algorithms, is not interesting in practice. The second fundamental change relies on the representation of candidate dosing regimens. While all previous work but [15] use an integer representation with discretised dose values, we use instead real-numbers for representing daily dosages. We argue that this encoding allows the exploration of a wider search space of possible dosing regimens. Moreover, this encoding prompted us to use an evolutionary algorithm specifically tailored to continuous optimisation (differential evolution [17,18]), rather than standard genetic algorithms. Our study also departs from previous work as we experiment with varied levels of bacterial resistance, coinfections with two strains of bacteria, and two ways of administering the drug: orally and intravenously. We contrast the optimised dosing regimens against the standard practice of fixed-daily doses with the same total amount of antibiotic. In summary, our study is guided by the following research questions.

How do optimised regimens vary according to the:

1. *antimicrobial resistance level?*
2. *form of administering antibiotics, orally vs. intravenously?*
3. *presence of a single bacterial strain vs. two strains of bacteria with different resistance levels?*

2. Related work

Other group of authors have used genetic algorithms to optimise antibiotic dosing regimens. Cicchese et al. [12] use genetic algorithms and surrogate-assisted optimisation to design regimens to treat Tuberculosis infections. Their formulation assumes that doses are fixed across the treatment, and vary instead the frequency of application of multiple drugs. The single objective function has two terms measuring the average time to eradication, and the dose size and frequency of antibiotics. Treatments are evaluated using a hybrid, multiscale model that combines agent-based modelling with differential equations, and a pharmacokinetic model.

Colin et al. [14] use a genetic algorithm to optimise a dosing guideline for intermittent infusion of vancomycin in adults. They encode dosing regimens as combinations of discretised loading doses, maintenance doses and dosing intervals. Although the loading and maintenance doses can vary across candidate solutions, a given solution holds the same loading and maintenance dose with varying dosing intervals. The formulation uses a single objective function with several constraints, and only focuses on the pharmacokinetic model (antibiotic concentrations), without explicitly modelling the bacteria infection, to simulate an adult patient population.

Our formulation differs from previous work mainly in the type of encoding candidate dosing regimens. The underlying simulation model of bacterial infection and interaction with antibiotics is also different from the approaches described above. Moreover, we contrast two ways of administering the drug (orally vs. intravenously) and experiment with varied levels of bacterial resistance, as well as coinfections with two strains of bacteria.

3. Methodology

3.1. Biomedical background

Once an antibiotic is chosen, conventional treatments have three main characteristics: the concentration of each dose, the time interval between doses and the total number of doses given. These characteristics are usually decided by the manufacturer or a health body, and usually consist of fixed sized doses at fixed time intervals. For example, a course of *Amoxicillin* may be 250 mg taken 3 times daily for 5 to 7 days [19]. While these fixed-dose treatments may be effective, they may not be the optimal dose or duration to administer the antibiotic most efficaciously. Although we are taking a theoretical approach, our parameters are 'loosely' based around an *E. coli UTI* infection being treated with *Amoxicillin*. This is in comparison to some of our previous work, where parameters were more arbitrarily chosen [11,13,15].

Dose regimens are often based on Pharmacokinetics and Pharmacodynamics studies of target populations. One significant characteristic of the bacterial population is minimum inhibitory concentration (MIC). This is the lowest concentration (in μg/mL) of an antibiotic that inhibits the population growth of a given strain of bacteria. In this work, we have chosen four MIC values: sensitive, 8 μg/mL, intermediate, 16 μg/mL and 24 μg/mL, and resistant, 32 μg/mL [20].

For short-term infections, there is often only one type of bacteria present in the host, which is the case for most healthy people when they are suffering from a bacterial infection. Later in this study, we consider immunosuppressed hosts. Here, the body is more susceptible to infection, and this could result in the host having multiple bacterial infections at the same time. Alternatively, two strains can also be present when a mutation occurs to create a more antibiotic resistant one.

3.2. Mathematical model

The mathematical model used follows a similar formulation as in [10,11,13,15], where a population of bacteria is simulated with a Markov chain approach using the Gillespie algorithm [21], and the effect of an antibiotic treatment to eradicate the infection is considered as detailed below. The model simulates the bacteria population through the duration of treatment plus an extra 3 days to allow the antibiotic in the blood to dissipate and to establish if the bacteria population has reached the count of 0 (treatment is successful) or not (treatment is not successful, or failed).

A n-day treatment is denoted as a vector $\overline{x} = (x_1, x_2, \dots, x_n)$, where x_i represents the dosage taken on day i, with $x_i \geq 0$. In this formulation, x_i are real positive numbers, $x_i \in \mathbb{R}$. The maximum total antibiotic, $\sum_{i=1}^{n} x_i$, is selected based on the amount needed to cure the host with a fixed daily-dose regimen, for the specific MIC of the bacteria determined empirically using the model. The time interval between doses is fixed at 24 h throughout this study.

Antibiotic Modelling. Pharmacokinetics/Pharmacodynamics (PK/PD) modelling is the basis of modern-day pharmacotherapy. Pharmacokinetics describes the drug concentration over time inside the host, while Pharmacodynamics observes the effects of the drug on the infection and on the host. In other words, pharmacokinetics answers the question 'what the body does to the drug', while pharmacodynamics — 'what the drug does to the body' [22,23]. In our study, when the drug is first taken, it enters the stomach at concentration C_S; this concentration decreases as it moves from the digestive system into the blood, whereby they become effective in fighting the bacterial infection, with this concentration denoted as C_B. This process is modelled by equations in Table 2. Parameters a and g correspond to the degradation half-life of the time the antibiotic takes to be absorbed in the gastric juices and in the host's blood — 15 h in the gastric juices and half an hour in the

Algorithm 1: Outline of the stochastic mathematical model with one bacteria strain.

1: $treatment = \{x_1, x_2, \ldots x_n\}$
2: initial bacteria population $B = 700$
3: initial antibiotic concentrations $C_B = 0$, $C_S = 0$
4: $time = 0$ minutes
5: time step $\tau = 15$ minutes
6: $end\_of\_day = 1440$ minutes (= 24 hours)
7: $deadly\_level\_bacteria = 2000$
8: **for** day 1 until the last day of treatment + 3 extra days **do**
9: $C_S = C_S + treatment(day)$ (take dose)
10: **while** $time \leq end\_of\_day$ and $0 < B < deadly\_level\_bacteria$ **do**
11: calculate average number of bacteria created p_1 (Table 2)
12: calculate average number of bacteria deaths p_2 (Table 2)
13: update bacteria population: $B = B + P(\tau p_1) - P(\tau p_2)$
14: update time: $time = time + \tau$
15: update antibiotic concentrations, C_S and C_B (Table 2)
16: **if** $B \geq deadly\_level\_bacteria$ **then**
17: Treatment is unsuccessful
18: **end if**
19: **end while**
20: **end for**
21: **if** $B \leq 0$ **then**
22: Treatment is successful
23: **else**
24: Treatment is unsuccessful
25: **end if**

Table 1
Mathematical model parameter values and references.

Parameter	Description and reference	Value
a	Absorption rate of antibiotics in the stomach [24]	33.27 h^{-1}
g	Degradation rate of antibiotics in the blood [24]	1.11 h^{-1}
p	Proportion of antibiotics that reaches the blood [24]	95%
m	Immune system response rate [25]	0.1 h^{-1}
r	Replication rate of bacteria [26]	0.5 h^{-1}
b_1	Maximum kill rate of the antibiotic (as $C_B \to \infty$)	2.5 h^{-1}
b_2	Level of antibiotic giving half max kill rate [11]	1.5137 $h^{-1} \times mic$
mic	Min inhibitory concentration (MIC)	8,16,24 or 32 μg/mL
k	Hill coefficient in antibiotic induced death	4

blood [24]. As about 5% of the antibiotic is lost in the gastric juices, the values for p is set to 0.95. [24].

Bacteria modelling (one strain). Where a single type of bacteria is present in the host, there are two events that happen: birth of bacteria (p_1) and death of bacteria (p_2) shown in Table 2. In p_1, the term rB represents the bacteria's binary fission for the time step, producing exponential growth at rate r - this form is based on biological experiments carried in [10], where bacteria grew exponentially and the hosts (larvae) died before any slow down of bacteria growth occurred; hence, we omit a carrying capacity in this term, and instead include a host death threshold if the bacteria exceeds 2000. In p_2, we sum the natural death of the bacteria, due to the host's immune system m, and antibiotic induced death rate represented by parameters b_1, b_2, mic and k.

The pseudocode, for one-strain model, can be seen in Algorithm 1, with parameters presented in Table 1.

Bacteria modelling (two strains). When modelling two strains of bacteria, S denotes the bacterial strain with a lower MIC and is more susceptible to the antibiotic, while R denotes the bacterial strain with a higher MIC, being more resistant and requiring a higher dose of antibiotics. In this case, the mathematical equations in Table 2 are replaced by those in Table A.1. There are now five events that take place in the simulation: birth of new bacteria of each type (p_1 and p_2), death of each type of bacteria (p_3 and p_4) and finally p_5 representing the horizontal gene transfer process (resistance gene from the R bacteria strain is passed on to the S bacteria strain.

Table 2
List of all events for simulating the population of bacteria during treatment.

Events	Description	Bacteria population (B) change
p_1	Birth of new bacteria	rB
p_2	Death of bacteria	$mB + \frac{b_1 C_B^k}{C_B^k + b_2^k} B$
C_B	Concentration of antibiotics in the blood	$C_B + \tau(paC_S - gC_B)$
C_S	Concentration of antibiotics in the stomach	$C_S - \tau aC_S$

Implementation and technical set up. To speed up the simulation process, we use an approximation of the Gillespie algorithm, known as Tau-leaping [21]. Following preliminary model runs, we settled on a fixed time step of $\tau = 15$ minutes, and updated the number of bacteria using the equation in Algorithm 1 (line 14), where $P(\tau p_i)$ is a Poisson distributed random variable with mean τp_i.

Our implementation uses Python with the Numba JIT compiler [27] to parallelise the simulation runs on up to 32 computer cores, significantly speeding up the process.

Table 3
Values for the minimum inhibitory concentration (MIC), and maximum total antibiotic used.

MIC [μg/mL]		8	16	24	32
Total antibiotic [mg]	higher	150	300	450	600
	lower	125	250	400	550

Table 4
Differential evolution control parameter values.

Parameter	Description	Value
F	Scaling factor (mutation)	(0.7, 1)
P	Population size	150
Cr	Crossover rate	0.7

3.3. Computational optimisation

Problem formulation. The task at hand is formulated as an optimisation problem. Specifically, as a single objective minimisation problem with a single linear constraint. The objective to minimise is the failure rate f_r measured as the ratio of simulation runs, using the stochastic model described in Section 3.2, where the bacteria population is not eradicated, that is where the bacteria population size is above zero after three days of the last regimen dose. This can be interpreted as treatment failure. We used a number of 10 000 simulation runs, and the failure rate is the ratio of the number of runs where the bacterial population is eradicated out of the 10 000 runs.

The constraint accounts for a maximum total antibiotic allowed for treatment. The total antibiotic used by a regimen vector $\overline{x} = (x_1, x_2, \ldots, x_n)$, is simply the sum $\sum_{i=1}^{n} x_i$ of its daily doses. The maximum total antibiotic allowed is modelled as a hard constraint, which means that a regimen vector that exceeds the allowed maximum A_{total} is deemed invalid and thus discarded by the optimisation process. More formally, the optimisation problem can be stated as follows:

$$\begin{aligned} &\text{Find vector } \overline{x} = (x_1, x_2, \ldots, x_n),\ x_i \in \mathbb{R}^+ \\ &\text{to minimise function } f_r \\ &\text{subject to the constraint } \textstyle\sum_{i=1}^{n} x_i \leq A_{total} \end{aligned}$$

In our experiments the duration of treatment was set to 10 days, $n = 10$ and no upper bound is imposed on the daily doses. Table 3 reports the minimum inhibitory concentration (MIC) values used in our experiments. For each MIC value, two values for the total antibiotic constraint were considered, which we name here in relative terms *lower* and *higher*. The *higher* values were selected in such a way that the best fixed-dose treatments in simulation reach a failure rate below 1%. The *lower* values allow us to explore the impact on the failure rate of reducing the total amount of antibiotic for both for fixed-dose and optimised treatments.

Differential Evolution (DE) is a population based stochastic search method, designed to solve continuous optimisation problems, and able to handle non-differentiable, nonlinear and multimodal objective functions [17]. DE is amongst the state-of-the-art evolutionary algorithms for continuous optimisation, and has been successfully applied to a variety of problems in science and engineering [28]. The growing evidence supporting the excellent performance of DE in terms of accuracy, convergence speed and robustness, in domains including electronics, manufacturing, machine learning, bioinformatics and biomedical-engineering [18,28], makes it an ideal choice for our purposes.

A feature of DE, distinguishing it from other evolutionary algorithms, is its *differential* mutation operator. Given a population of candidate solutions in $\mathbb{R}^n$ a new mutant vector $\overline{x}'$ is produced by adding a perturbation vector to an existing one, $\overline{x}' = \overline{x} + \overline{p}$, where the perturbation vector $\overline{p}$ is the scaled vector difference of two other, randomly chose population members $\overline{p} = F \times (\overline{y} - \overline{z})$. The other reproduction operator is uniform crossover, subject to a crossover rate parameter $Cr \in [0, 1]$. In general, a DE algorithm has three control parameters, the scaling or mutation factor F, the population size P and the crossover rate Cr. Table 4 reports the DE control parameter values used in our experiments. Our experiments use dithering for the mutation factor F, as it can help the speed of convergence. Dithering uniformly at random (from a given tuple (min, max)) changes the mutation constant on a generation by generation basis. Over the years, several DE variants have been proposed [18]. Here we use the classic 'rand/1/bin' strategy, where 'rand' indicates that base vectors are randomly chosen, '1' means that only one vector difference is used to form the mutated population, and the term 'bin' (from binomial distribution) indicates that uniform crossover is employed when creating the trial population.

Stopping condition. The stopping condition for the DE runs was set as a maximum number of iterations. We used a maximum of 4 000 iterations for experiments with MIC = 8 μg/mL, 8 000 iterations for MIC = 16 and MIC = 24 μg/mL, and 10 000 iterations for MIC = 32 μg/mL. We needed to scale up the iterations with the MIC value as a larger amount of antibiotic was required the higher the MIC, which resulted in an increased feasible search space. Fig. 1 shows typical DE failure rate convergence profiles. The failure rate appears to stabilise (within a margin of error) well before the chosen iteration bounds.

Constraint Handling. A common way of handling constraints within evolutionary algorithms is to apply penalty functions. In its simplest form, the function to be minimised can be computed by penalising the objective function with a weighted sum of constraint violations. A disadvantage of this approach, however, is that one or more additional penalty parameters are expected to be set by the user a priory, which requires additional effort. Therefore, in our experiments we adopted the constraint handling technique proposed in [29], where the replacement rule of the DE algorithm is modified. Specifically, when compared with the corresponding member in the population, a trial (mutant) vector will be selected if: (i) it is feasible and provides a lower or equal objective function value, (ii) it is feasible while the current vector is unfeasible, or (iii) it is infeasible but provides a lower or equal constraint violation. This method has the advantage in our formulation of not requiring any additional parameter value other than the total antibiotic (A_{total}) constraint value.

Implementation and technical set up. The optimisation process was implemented in Python using NumPy [30] and the Differential Evolution algorithm with its associated constraint handling methods available in SciPy [31]. A total of 10 DE runs were conducted for each MIC and A_{total} constraint values.

Re-evaluation of best-found solutions. As the underlying mathematical model of bacterial infection is stochastic, the evaluation of the failure rate f_r during DE runs is susceptible to noise. This is due to both the mathematical model using a fixed number of 10 000 simulation runs, and the greedy DE selection bias, where noise could produce optimistic estimates of the failure rate. To counter these inaccuracies, all final solutions are re-evaluated by running the stochastic model 1 000 000 times. Binomial confidence intervals (with 95% confidence limit) are then calculated for each of the failure rates f_r, and these confidence intervals are used when comparing solutions in order to establish which one truly performs best.

4. Results

Our results are organised into 3 subsections, reporting experiments with a single strain of bacteria and antibiotic administered orally Section 4.1, a single strain of bacteria and antibiotic administered intravenously Section 4.2, and two strains of bacteria and antibiotic administered orally Section 4.3. For all experiments, the antibiotic is administered at fixed 24-hour intervals for the duration of treatment.

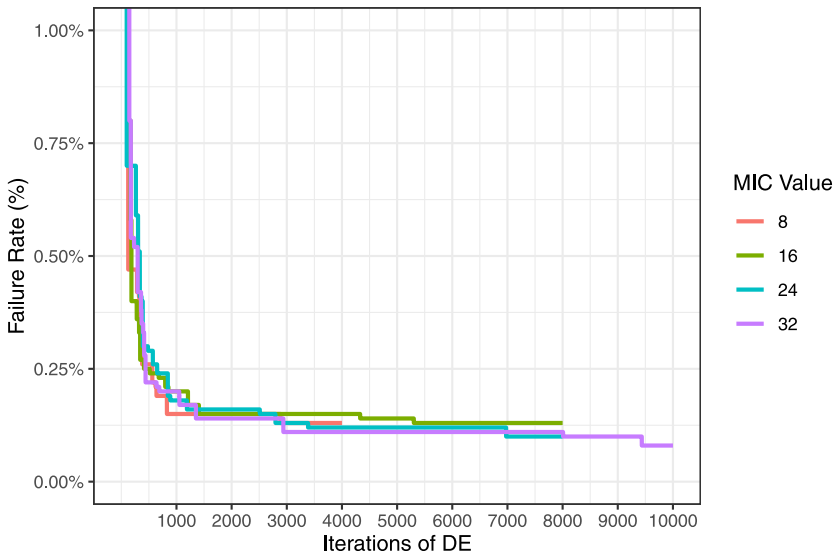

Fig. 1. Failure rate convergence over DE iterations for experiments with different MIC values and total antibiotic constraint A_{total} as indicated in Table 3.

4.1. Single bacteria strain and oral administration

We start by contrasting the effectiveness of fixed-dose treatments against those optimised by DE. In order to identify the fixed-dose benchmarks, we compute the failure rates, using the mathematical model, of fixed-dose treatments with duration between 5 and 10 days, for all the MIC and total antibiotic constraint values. The daily doses of fixed-dose treatments are simply the total antibiotic values divided by the number of treatment days. The resulting failure rates are plotted in Fig. 2. We have re-scaled the plot to only include treatments with $f_r \leq 8\%$ so some data points are missing from the last two plots.

We observe that treatments of length 6, 7 and 8 days provide the lowest failure rate f_r, while 9 and 10 days regimens produce the highest failure rates f_r, especially for bacteria with *MIC* 8 and 16 μg/mL which aligns with the clinical experiences of antibiotic courses of 7 days being the most commonly prescribed.

To compare the best fixed-dose treatments against the DE-optimised treatments, we completed 10 runs of DE for every combination of MIC and total antibiotic. The resulting scatter plot of failure rates can be seen in Fig. 3. For comparison, the figure also shows the best fixed-dose failure rates (taken from Fig. 2) as a black coloured marker. Note that the failure rates f_r of all optimised regimens are based on re-evaluating the mathematical model 1 000 000 times. In addition, the dose was set to 0 for each day when a DE-optimised treatment recommended a dose of less than 5 mg as doses under 5 mg have little effect on the success of the treatment

We observe that the fixed-dose treatments are less effective, that is, have a higher failure rate f_r than any of the optimised treatments. For the treatments where the MIC is 8 and 16 we can see that even a small increase in the total antibiotic results in an improvement from around $f_r = 2.25\%$ to $f_r = 0.3\%$, whereas with MIC at 24 and 32, the improvement is slightly less. This is expected, as a higher MIC requires more antibiotic to kill the bacteria, so adding 50 mg of antibiotic amounts to a lesser relative increase of the total amount of antibiotic.

4.1.1. Dosage profile of optimised treatments

Fig. 4 plots the dosage profiles of the three best optimised treatments. The best treatments are those with lowest failure rates after re-evaluation. For comparison, the (constant) dosage profile of the best fixed-dose treatment (as determined in Fig. 2) is also shown in black. Failure rates and confidence intervals are listed on the right-hand side of each plot.

Across all experiments, we observe that the failure rates of the best optimised treatments are approximately between 20% and 35% lower than the failure rate of the corresponding best fixed-dose treatment. The failure rate reduction appears to diminish with higher MIC and higher total antibiotic values. For instance, the lowest failure rate reduction of 21% is found for the experiment with MIC = 32 μg/mL and the higher total antibiotic constraint of 600 mg (Fig. 4*(h)*).

None of the experiments produce a clear *best* optimised treatment, as the confidence intervals of several optimised treatments overlap. In addition to being virtually indistinguishable by failure rate, the dosage profiles of the three best optimised treatments appear to follow a similar pattern.

- All optimised treatments for a given MIC value and antibiotic constraint agree on the treatment duration. In most cases, this is the same as the length of the corresponding fixed-dose treatment (except for the experiments with MIC = 8 or 16 μg/mL and the lower antibiotic constraint, where optimised treatments take one day longer).
- All optimised treatments start with a high dose on the first day, followed by $n-2$ doses that are roughly similar to the corresponding fixed-dose treatment, and tapering off with a lower dose on the final day, where n is the duration of the treatment. The first and last doses vary across experiments. In most cases, the first dose is approximately 150% of the second dose, and the final dose is about 50% of the second dose.

4.1.2. Distribution of time to clear infections

In addition to treatment failure rates, we investigate the *time to clear the infection* of successful treatments by counting the number of days it takes for the bacterial population to drop to zero. Fig. 5 plots the distributions of the time to clear for each experiment, both for the best fixed-dose treatment (the left-most column of each plot) and for the three best optimised treatments. Distributions are presented as colour-coded columns, where the height of each colour block corresponds to the number of hosts (out of 1 000 000) that cleared the infection on the given day of the treatment. Shown on top of each column is the *expected time to clear* the infection in days. Note that even though treatments are at most 8 days long, hosts may clear the infection after the last day of treatment. Failed treatments, that is, cases where the infection is not cleared within 13 days, are excluded from the distributions.

Across most experiments, we observe that optimised treatments clear infections faster. In particular, most optimised treatments clear significantly more infections on or before day 4 than the corresponding

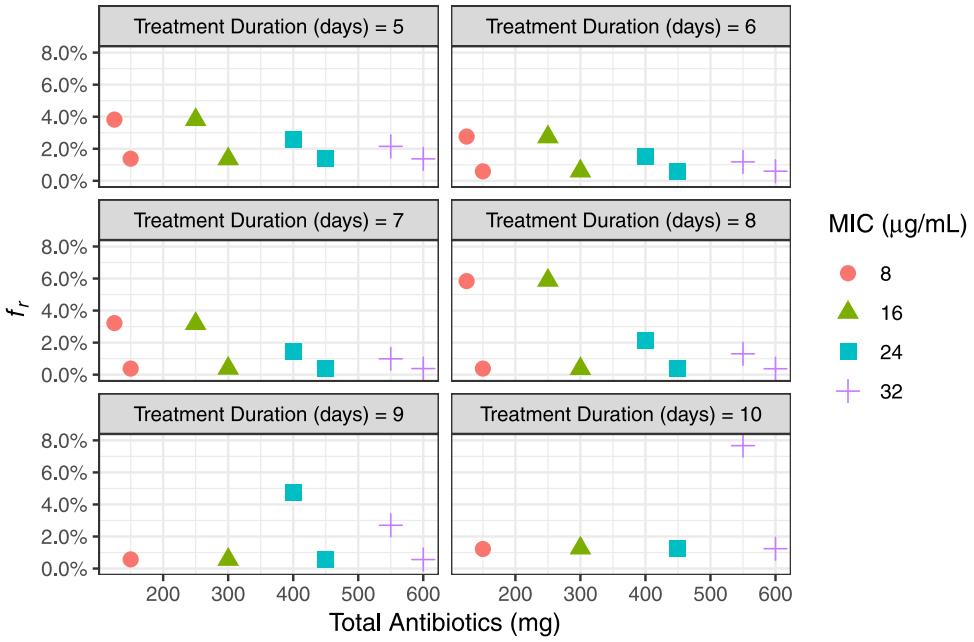

Fig. 2. Treatments with fixed daily doses by *MIC* value, maximum total antibiotic and length. The plot has been re-scaled so only solutions with $f_r \leq 8\%$ are presented.

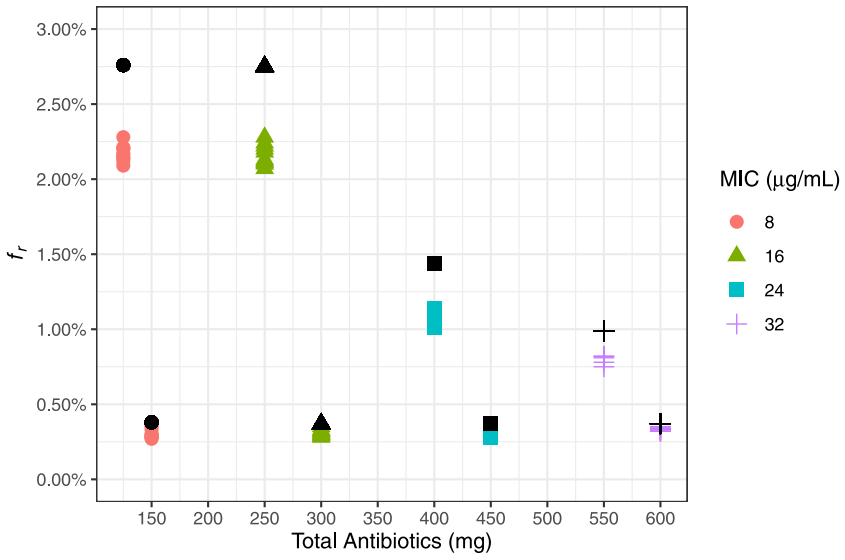

Fig. 3. Optimised treatments with different daily doses by *MIC* value and maximum total antibiotics. The black markers represent the fixed treatment with the lowest failure rate f_r for that configuration.

fixed-dose treatments, resulting in a reduction of the expected time to clear by between 0.4 and 0.8 days. The exception are the experiments with MIC levels 8 and 16 µg/mL and lower total antibiotic constraint, where the distributions of time to clear of the optimised treatments are very similar to the distributions of the corresponding fixed dose treatments. However, the optimised treatments in these two experiments are one day longer than the fixed dose treatments, which explains why we do not observe improvements in the time to clear infections in these cases. (Note that we were only optimising the failure rate of treatments, not the time to clear infections.)

4.1.3. Discussion

Our results suggest an optimal treatment duration of 7 days if optimised against the lower total antibiotics constraint, and 8 days if optimised against the higher constraint. This is broadly in line with clinical practice, where most of the treatments prescribed are 5 or 7 days long.

We find that some optimised treatments are slightly longer than the fixed-dose ones, but perform better. We see a bigger improvement of failure rate when the MIC levels of the bacteria are at susceptible and intermediate resistance levels (8 and 16 µg/mL) to the antibiotics than when they are more resistant (24 and 32 µg/mL). As these are the majority of bacterial infections in hospitals, the optimised treatments would reduce the number of cases where bacteria survive after the end of the treatment, thereby reducing the risk of resistant strains emerging.

We also observe that optimised treatments clear infections faster than the corresponding fixed-dose treatments. This effect appears stronger when MIC levels are at the more resistant end (24 and 32 µg/mL). Thus, optimised treatments confer a second advantage, particularly for infections with resistant bacteria, by helping more patients recover quickly, thereby potentially reducing the burden on hospitals.

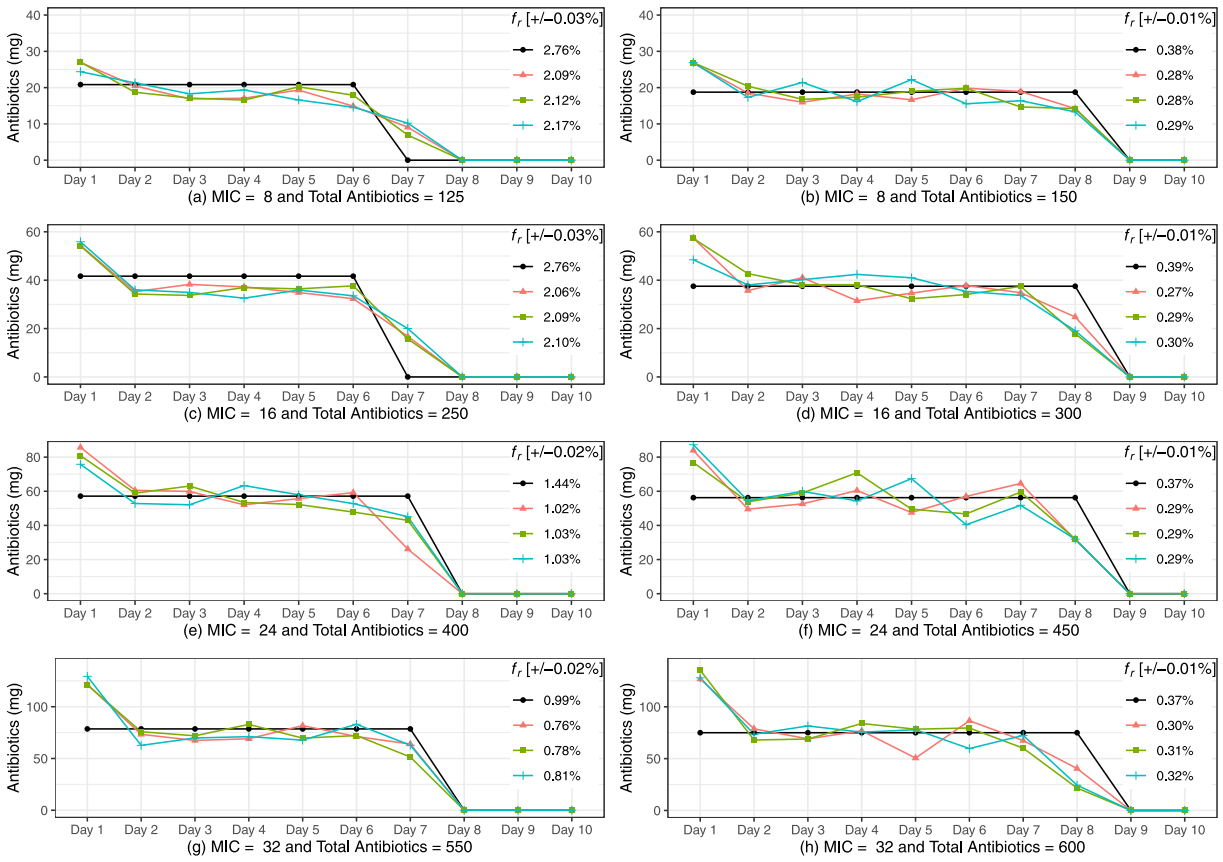

Fig. 4. Comparison of the dosage profile of the three best optimised treatments against the best fixed-dose treatment (coloured in black and shaped with a circle). Failure rates f_r are listed on the right-hand side, with confidence intervals in square brackets.

We attribute both the improvements in the failure rate and in the time to clear infections to the higher first-day dose of optimised treatments.

4.2. Single bacteria strain and intravenous administration

When antibiotics are injected intravenously, they go directly into the blood stream rather than through the stomach as in the previous Section 4.1. In terms of the mathematical model (described in Section 3.2) C_S, the concentration of antibiotics in the stomach, is set to zero, and p, the proportion of antibiotics that reaches the blood, is set to one (instead of the previous value of 0.95). In order to keep results comparable, the experiments reported here explore the same combinations of MIC value and total antibiotics constraint as in Section 4.1.

The scatter plot in Fig. 6 compares the failure rates of the best fixed-dose treatments against ten DE-optimised treatments. We observe a similar picture as for orally administered treatments, that is, DE-optimised treatments tend to have lower failure rates, and the difference between lower and higher total antibiotics constraint diminishes the higher the MIC value. We also observe that failure rates are lower than for the respective orally administered treatments in Fig. 3, which confirms that administering the drug intravenously increases effectiveness.

Fig. 7 plots the dosage profiles of the three best optimised treatments. For comparison, the profile of the best fixed-dose treatment is also shown in black colour. Failure rates and confidence intervals are listed on the right-hand side of each plot.

Across experiments with the lower total antibiotics constraint, we observe that the failure rates of the best optimised treatments are approximately between 15% and 30% lower than the failure rate of the corresponding best fixed-dose treatment. However, we see almost no improvement in failure rates for experiments with the higher total antibiotics constraint. In fact, the confidence intervals of many of the DE-optimised treatments overlap the confidence interval of the best fixed-dose treatment.

The general shape of the dosage profiles is similar to the shape of the orally administered treatments: a high first dose, followed by roughly constant doses, and tapering off on the final day. However, more than half of the optimised treatments are a day longer than the best fixed-dose treatment. In contrast, most orally administered treatments matched the fixed-dose treatment in duration.

4.2.1. Discussion

As observed in Fig. 7, DE barely manages to improve on the failure rate of the fixed-dose treatment in experiments with the higher total antibiotics constraint. There are two hypotheses for this — due to the stochastic nature of the mathematical model, the fitness function might be too noisy, making it difficult for the DE to find the optimal solution; or the fixed-dose failure rate is already near the optimal value.

To check whether the failure of DE in finding better solutions is down to the noisy fitness function, we performed additional experiments, increasing the number of runs of the mathematical model from 10 000 to 100 000, thereby reducing the noise on the fitness function by an order of magnitude (yet increasing the computational cost by an order of magnitude). The best failure rates found in these experiments ranged from $0.17\% \pm 0.01\%$ to $0.19\% \pm 0.01\%$. This is an improvement on the best fixed-dose failure rates of $0.22\% \pm 0.01\%$, although the relative

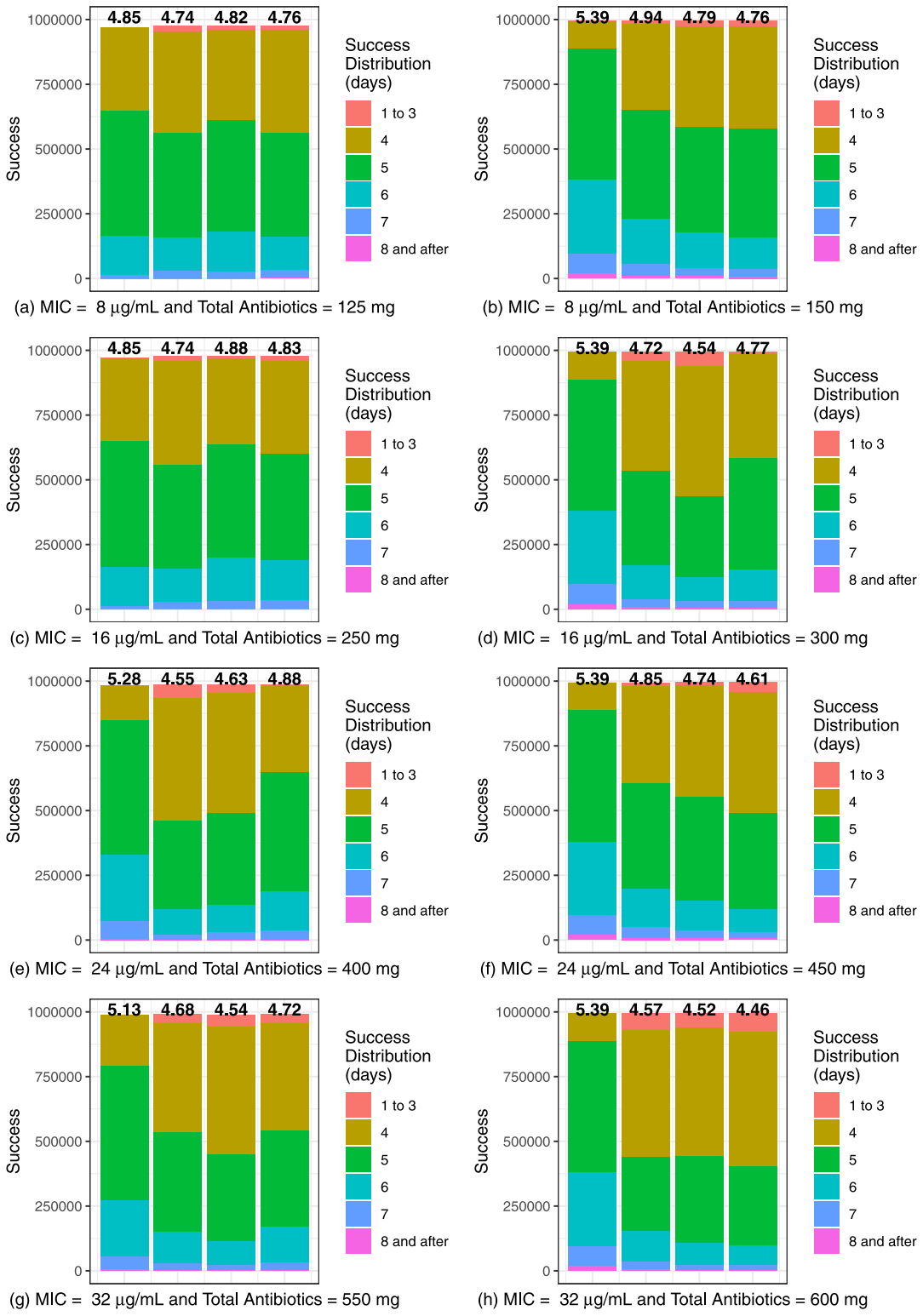

Fig. 5. Distributions of the time to clear the infection, comparing the best fixed-dose treatment (left-most column of each plot) to the three best optimised treatments. Expected time to clear is shown on top of each column. Each treatment was evaluated 1 000 000 times.

improvement of about 15% to 20% is smaller than observed in other experiments.

The additional experiments suggest that noise on the fitness function may prevent DE from converging to the optimum. However, the modest improvements despite reducing the noise by an order of magnitude also suggest that there is not a single optimal treatment but a wide basin of treatments with very similar near-optimal failure rates.

4.3. Two bacteria strains with oral administration

In cases when people are immunocompromised, it is common that they could carry multiple types of bacteria or several strains of the same type of bacteria. In this set of results, we are modelling the case when 95% of the bacterial population have a resistance of MIC = 8 μg/mL and

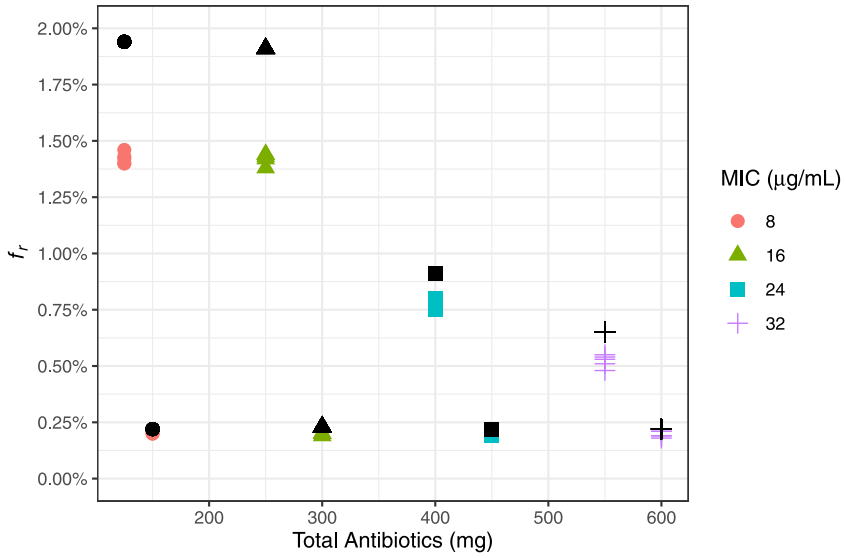

Fig. 6. Optimised treatments administered intravenously by *MIC* value and maximum total antibiotics. The black markers represent the fixed-dose treatment with the lowest failure rate f_r for that configuration.

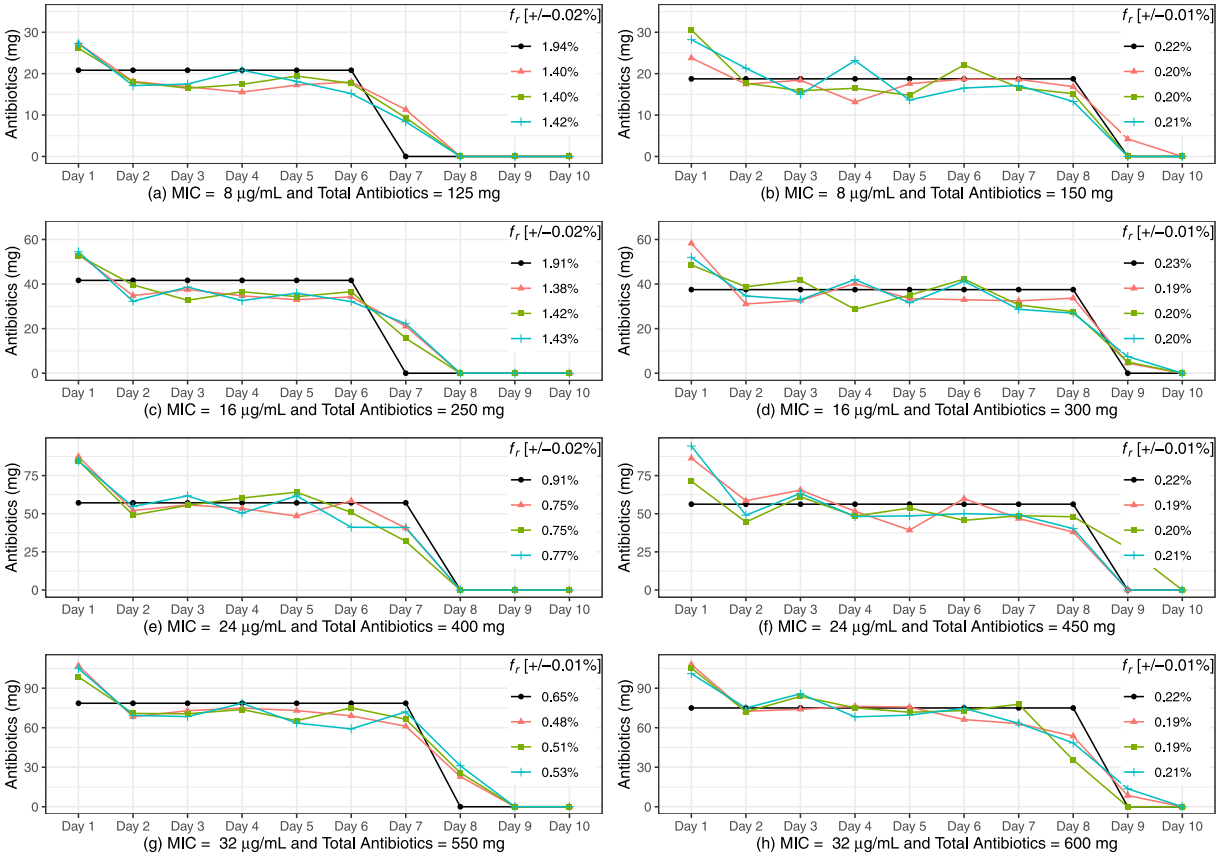

Fig. 7. Comparison of the dosage profile of the three best optimised intravenous treatments against the best fixed-dose treatment (in black). Failure rates f_r are listed on the right-hand side, with confidence intervals in square brackets.

5% of the bacterial population make up a strain with a more resistant MIC (16, 24 or 32 µg/mL). Antibiotics are administered orally.

We first examined treatments using the same total amount of antibiotics as in the experiments with only one strain of bacteria in Section 4.1. We observed failure rates around 10% when the more resistant strain has MIC = 16 µg/mL. However, the failure rates rise to an average of 97% for MIC = 24 µg/mL, and to 100% for MIC = 32 µg/mL. This shows that if a patient is treated for less resistant bacteria when a more resistant strain is present even in small amounts, the treatment will likely fail.

It seems plausible that treating a multi-strain infection will require more antibiotics than would be required for the less susceptible strain on its own, and less than would be required to treat the most resistant strain on its own. This is confirmed by the findings in Fig. 8, which plots failure rates for combinations of the total antibiotics constraint and the MIC value of the more resistant strain. We are showing the

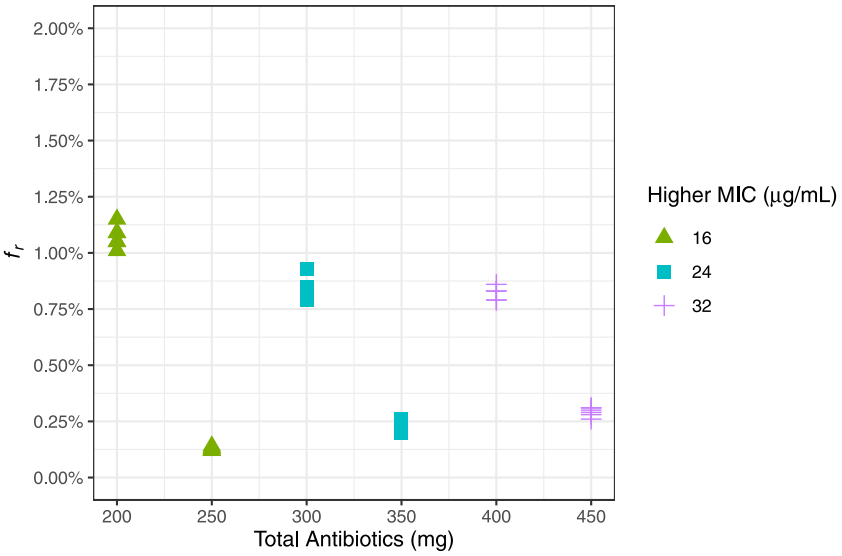

Fig. 8. Treatments with $f_r \leq 1\%$ when there are two strains of bacteria — one with MIC = 8 µg/mL that makes 95% of the initial bacteria population and one that corresponds to the MIC shown in the plot that makes 5% of the initial bacteria population.

result of five runs of the DE with 10 000 iterations for each scenario. For each MIC value, the figure shows two treatments that differ by 50 mg in the total amount of antibiotics used. The treatments with the lower total antibiotics constraint are 7 days long, the treatments with the higher constraint are 8 days. The shape of the treatments is not shown but follows the same pattern we observed before — high first dose, roughly constant middle doses, and tapering off with a smaller final dose.

4.3.1. Discussion

We saw that even if a small percentage of the bacteria population develops a mutation increasing resistance, the treatment could become unsuccessful. Thus, more antibiotics are needed where a multi-strain infection is suspected. In our model, we found that the amount of antibiotics needed to guarantee a failure rate well below 1% is quite close to the amount of antibiotics required to treat that the more resistant strain on its own — even though that more resistant strain makes up only 5% of the initial bacteria population.

4.4. Extrapolating the optimised regimens

As observed in Figs. 4 and 7, the shape of optimised treatments always follows the same pattern: a high first dose, followed by roughly constant doses, and tapering off on the final day. The ratio of first to second doses varies across experiments but is often close to 1.5. This leads us to extrapolate the following simple way of formulating an optimised treatment without running the DE algorithm.

Suppose the standard fixed-dose regime is a daily dose of $2 \times x$ mg over n days. Then the extrapolated optimised dose regime consists of a first dose of $3 \times x$ mg, followed by $n-2$ doses of $2 \times x$ mg, followed by a final dose of x mg. Table 5 contrasts the failure rates of the best DE-optimised treatments (orally administered) against the failure rates of treatments of the same length but using the extrapolated dosage regime. This shows a slight further improvement (around 5 to 10%) of failure rates across the board. (We see a similar improvement for intravenously administered drugs.)

The extrapolated dose regime could easily be implemented in a real-life scenario where x mg is the dosage of a single pill, and the patient takes 3 pills on the first day of treatment, followed by $n-2$ days of 2 pill doses, and 1 pill on the final day. Our modelling predicts that such a dose regime would significantly improve treatment failure rates compared to the standard fixed dose treatment. However, these predictions need to be clinically validated.

Table 5
Comparison of failures rates f_r of the best fixed-dose treatments, the best DE-optimised treatments, and the corresponding extrapolated treatments.

MIC [µg/mL]	8	16	24	32
Total antibiotics [mg]	125	250	400	550
treatment length [days]	7	7	7	7
f_r fixed dose [%]	2.76 ± 0.03	2.74 ± 0.03	1.41 ± 0.02	1.00 ± 0.02
f_r best DE [%]	2.09 ± 0.03	2.06 ± 0.03	1.02 ± 0.02	0.76 ± 0.02
f_r extrapolated [%]	2.00 ± 0.03	1.98 ± 0.03	0.98 ± 0.02	0.71 ± 0.02
MIC [µg/mL]	8	16	24	32
Total antibiotics [mg]	150	300	450	600
Treatment length [days]	8	8	8	8
f_r fixed dose [%]	0.36 ± 0.01	0.38 ± 0.01	0.38 ± 0.01	0.38 ± 0.01
f_r best DE [%]	0.28 ± 0.01	0.27 ± 0.01	0.29 ± 0.01	0.30 ± 0.01
f_r extrapolated [%]	0.25 ± 0.01	0.25 ± 0.01	0.25 ± 0.01	0.25 ± 0.01

5. Conclusion

Antimicrobial resistance is a growing global threat to healthcare and food production. To deal with this complex challenge, a range of approaches are required, critically including novel strategies to optimise the use of existing antibiotics. This study uses mathematical modelling and state-of-the-art evolutionary algorithms for optimising dosing regimes tailored to bacterial infections with different level of resistance. We also explored two forms of administering antibiotics (orally and intravenously), as well as infections with a single strain and two strains of bacteria. Our formulation encodes dosing regimens as vectors of real numbers and uses a linear constraint on the total antibiotic used.

Our main goal was to design optimised regimens with lower failure rate than the standard fixed-daily dose regimens for the same amount of antibiotic. The resulting optimised regimes have varying daily doses and achieve an improved lower failure rate of between 20% and 35% when compared to fixed-dose regimens with the same amount of drug. All optimised regimens, for n days in duration, start with a high dose on the first day, followed by $n-2$ doses that are roughly similar to the corresponding fixed-dose regimen, and tapering off with a lower dose on the final day. The first and last doses vary across experiments. In most cases, the first dose is approximately 150% of the second dose, and the final dose is about 50% of the second dose.

A general pattern can thus be extrapolated of how treatments could be optimised, where the first dose is $3 \times x$ mg, followed by $2 \times x$ mg

Table A.1
List of all events for simulating the population of bacteria during treatment where two strains of bacteria are present. Here, S bacteria is more susceptible with a lower MIC and R is more resistant with a higher MIC.

Events	Description	Bacteria population (S, R) change
p_1	Birth of new S bacteria strain	$r_S S$
p_2	Birth of new R bacteria strain	$r_R R$
p_3	Death of S bacteria	$m_S S + \frac{b_{S1} C_b^{k_S}}{C_b^{k_S} + b_{S2}^{k_S}} S$
p_4	Death of R bacteria	$m_R R + \frac{b_{R1} C_b^{k_R}}{C_b^{k_R} + b_{R2}^{k_R}} R$
p_5	S bacteria becomes R due to the horizontal gene transfer process	$\theta S R$

and last dose of x mg, where $2 \times x$ mg is the standard daily fixed dose currently prescribed. It is important to note that different antibiotics have different levels of toxicity, however, taking an extra dose of antibiotics is unlikely to cause serious harm. It is important to note that clinical validation is needed, however that is beyond the scope of this study.

As an interdisciplinary project, future work can follow several directions. First, a real-world validation of the findings of this study would be desirable. Second, the underlying mathematical model can be extended to incorporate, for example, patient attributes and regimens with multiple drugs. Finally, additional state-of-the-art computational techniques could be incorporated for managing noise, constraints, multiple objectives, and reducing the computational running times.

Appendix. Two strains of bacteria — Table

See Table A.1.

References

[1] Brunton LA, Desbois AP, Garza M, Wieland B, Mohan CV, Häsler B, et al. Identifying hotspots for antibiotic resistance emergence and selection, and elucidating pathways to human exposure: Application of a systems-thinking approach to aquaculture systems. Sci Total Environ 2019;687:1344–56.

[2] Courtenay M, Castro-Sanchez E, Fitzpatrick M, Gallagher R, Lim R, Morris G. Tackling antimicrobial resistance 2019–2024–the UK's five-year national action plan. J Hosp Infect 2019;101(4):426–7.

[3] Spellberg B, Bartlett JG, Gilbert DN. The future of antibiotics and resistance. N Engl J Med 2013;368(4):299–302.

[4] World Health Organization, et al. No time to wait: Securing the future from drug-resistant infections. 2019.

[5] Luz CF, van Niekerk JM, Keizer J, Beerlage-de Jong N, Braakman-Jansen LA, Stein A, Sinha B, et al. Mapping twenty years of antimicrobial resistance research trends. Artif Intell Med 2022;123:102216. http://dx.doi.org/10.1016/j.artmed.2021.102216.

[6] Charani E, McKee M, Ahmad R, et al. Optimising antimicrobial use in humans – review of current evidence and an interdisciplinary consensus on key priorities for research. Lancet Reg. Health - Europe 2021;7:100161.

[7] Rawson TM, Wilson RC, O'Hare D, Herrero P, Kambugu A, Lamorde M, et al. Optimizing antimicrobial use: Challenges, advances and opportunities. Nat Rev Microbiol 2021;1–12.

[8] McFarland LV. Alternative treatments for clostridium difficile disease: What really works? J Med Microbiol 2005;54:101–11.

[9] McKenzie C. Antibiotic dosing in critical illness. J Antimicrob Chemother 2011;66(2):ii25–31. http://dx.doi.org/10.1093/jac/dkq516.

[10] Hoyle A, Cairns D, Paterson I, McMillan S, Ochoa G, Desbois AP. Optimising efficacy of antibiotics against systemic infection by varying dosage quantities and times. PLoS Comput Biol 2020;16(8):e1008037.

[11] Paterson IK, Hoyle A, Ochoa G, Baker-Austin C, Taylor NG. Optimising antibiotic usage to treat bacterial infections. Sci Rep 2016;6:37853.

[12] Cicchese JM, Pienaar E, Kirschner DE, Linderman JJ. Applying optimization algorithms to tuberculosis antibiotic treatment regimens. Cell Mol Bioeng 2017;10(6):523–35.

[13] Ochoa G, Christie LA, Brownlee AE, Hoyle A. Multi-objective evolutionary design of antibiotic treatments. Artif Intell Med 2020;102:101759.

[14] Colin PJ, Eleveld DJ, Thomson AH. Genetic algorithms as a tool for dosing guideline optimization: Application to intermittent infusion dosing for vancomycin in adults. CPT: Pharmacomet Syst Pharmacol 2020;9(5):294–302.

[15] Goranova M, Contreras-Cruz MA, Hoyle A, Ochoa G. Optimising antibiotic treatments with multi-objective population-based algorithms. In: 2020 IEEE congress on evolutionary computation. IEEE; 2020, p. 1–7. http://dx.doi.org/10.1109/CEC48606.2020.9185489.

[16] Deb K, Pratap A, Agarwal S, Meyarivan T. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 2002;6(2):182–97.

[17] Storn R, Price K. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 1997;11(4):341–59.

[18] Das S, Mullick SS, Suganthan P. Recent advances in differential evolution – an updated survey. Swarm Evol Comput 2016;27:1–30. http://dx.doi.org/10.1016/j.swevo.2016.01.004.

[19] National Institute for Health and Care Excellence. Summary of antimicrobial prescribing guidance – managing common infections. 2020.

[20] IDEXX Laboratories, Inc. Microbiology guide to interpreting minimum inhibitory concentration (MIC). 2019, URL https://www.idexx.com/files/microbiology-guide-interpreting-mic.pdf.

[21] Gillespie DT. Approximate accelerated stochastic simulation of chemically reacting systems. J Chem Phys 2001;115(4):1716–33.

[22] Levison ME, Levison JH. Pharmacokinetics and pharmacodynamics of antibacterial agents. Infect Dis Clin 2009;23(4):791–815.

[23] Meibohm B, Derendorf H. Basic concepts of pharmacokinetic/pharmacodynamic (PK/PD) modelling. Int J Clin Pharmacol Ther 1997;35(10):401–13.

[24] Erah P, Goddard A, Barrett D, Shaw P, Spiller R. The stability of amoxycillin, clarithromycin and metronidazole in gastric juice: Relevance to the treatment of helicobacter pylori infection. J Antimicrob Chemother 1997;39(1):5–12.

[25] Handel A, Margolis E, Levin BR. Exploring the role of the immune response in preventing antibiotic resistance. J Theoret Biol 2009;256(4):655–62.

[26] Wang P, Robert L, Pelletier J, Dang WL, Taddei F, Wright A, et al. Robust growth of Escherichia coli. Curr Biol 2010;20(12):1099–103.

[27] Lam SK, Pitrou A, Seibert S. Numba: A llvm-based python jit compiler. In: Proceedings of the second workshop on the LLVM compiler infrastructure in HPC. 2015, p. 1–6.

[28] Das S, Suganthan PN. Differential evolution: A survey of the state-of-the-art. IEEE Trans Evol Comput 2010;15(1):4–31.

[29] Lampinen J. A constraint handling approach for the differential evolution algorithm. In: Congress on evolutionary computation, Vol. 2. 2002, p. 1468–73. http://dx.doi.org/10.1109/CEC.2002.1004459.

[30] Harris CR, Millman KJ, van der Walt SJ, et al. Array programming with NumPy. Nature 2020;585(7825):357–62. http://dx.doi.org/10.1038/s41586-020-2649-2.

[31] Virtanen P, Gommers R, Oliphant T, et al. Scipy 1.0: fundamental algorithms for scientific computing in Python. Nature Methods 2020;17:261–72. http://dx.doi.org/10.1038/s41592-019-0686-2.