

Simulation modelling for immunologists

Andreas Handel ^{1*}, Nicole L. La Gruta ² and Paul G. Thomas ³

Abstract | The immune system is inordinately complex with many interacting components determining overall outcomes. Mathematical and computational modelling provides a useful way in which the various contributions of different immunological components can be probed in an integrated manner. Here, we provide an introductory overview and review of mechanistic simulation models. We start out by briefly defining these types of models and contrasting them to other model types that are relevant to the field of immunology. We follow with a few specific examples and then review the different ways one can use such models to answer immunological questions. While our examples focus on immune responses to infection, the overall ideas and descriptions of model uses can be applied to any area of immunology.

Phenomenological models

Mathematical and statistical models used to extract patterns and information from data, without trying to specify detailed mechanisms that lead to the observed data.

Machine learning or deep learning

Data analysis approaches that generally use complex computational models to find patterns in the data, usually with the goal of making predictions.

¹Department of Epidemiology and Biostatistics, Health Informatics Institute and Center for the Ecology of Infectious Diseases, University of Georgia, Athens, GA, USA.

²Department of Biochemistry and Molecular Biology, Monash Biomedicine Discovery Institute, Monash University, Clayton, Victoria, Australia.

³Department of Immunology, St Jude Children's Research Hospital, Memphis, TN, USA.

*e-mail: ahandel@uga.edu

<https://doi.org/10.1038/s41577-019-0235-3>

The reductionist approach to science will continue to play an important role in scientific inquiry and progress. However, there is increasing interest across disciplines in studying the multiple interacting components of a given system simultaneously. This stems from the realization that it is often the complex interactions among these components that determine the ultimate outcome. While complexity in immunology is readily acknowledged, in many cases the ability to measure the contribution of each component is challenging. Even where this can be achieved, it can be difficult to interpret the results by looking at individual components independently. Mathematical and computational modelling can provide valuable information on the relative importance of different immunological components, how they are influenced by other components and how these relationships may vary across conditions. Models can provide informed hypotheses for experimental testing, generate a comprehensive map of the integrated performance of the immune system and identify potential targets for clinical manipulation of the immune response.

In this Review, we provide an introduction and overview of one category of models: those based on mechanistic simulations of an underlying system of interest. Our primary goal is to familiarize immunologists with, and increase interest in, these kinds of models, and to provide enough information for readers to understand the strengths, weaknesses and uses of such models. For those interested in pursuing modelling further, we also provide some pointers for getting started with developing and using simulation models.

Types of models

Models are everywhere in science. Models can be conceptual, including verbal models, graphs or charts; can be experimental, such as a specific cell culture system or mouse strain; or may take the form of quantitative

mathematical or computer models. Here we exclusively focus on the latter type of models.

The most common types of quantitative models are what we term phenomenological models (we avoid the often-applied term 'statistical' because the contrasting mechanistic models we focus on here can also be used in a statistical manner). Phenomenological models are applied to extract patterns or, more broadly, information from data. As such, whenever data are being analysed in some mathematical manner, this type of model is in play. Computing a correlation coefficient between two quantities of interest is an example of a very simple model that tries to detect a pattern. Regression models, in which a mathematical function is specified and the distance between the data and the function is minimized, also fall into this category¹. In recent years, the increase of available data has led to greater use of more complex phenomenological models, which increasingly go by names such as machine learning or deep learning approaches^{2,3}. These models do not explicitly describe the mechanisms by which patterns arise, and this is both a strength and a weakness. On the plus side, one can determine correlations, find patterns, deduce potential causation and make predictions without having to understand the underlying mechanisms governing a given system. The drawback is that such models provide, at best, hints regarding potential mechanisms. Especially for complex systems, one can often find correlations or patterns in the data without knowing whether these are indicative of mechanistic or causal connections (for some illuminating and entertaining examples, such as the strong correlation between divorces in Maine and the consumption of margarine, see REF.⁴).

To study mechanisms and processes, mechanistic models are ideal. One prominent type of such models uses simplified *in silico* representations of the processes underlying a system of interest to perform computer simulations.

Mechanistic models

Models that explicitly include specific mechanisms/processes governing the system of interest (usually in a very simplified way) in order to allow direct investigation of such mechanisms.

Simulation

Execution of a model on a computer; this often involves tracking changes of the system over time.

Other terms used for such models are systems models, dynamical models or simply mathematical models. In particular, the terms systems thinking and systems modelling have become popular in various fields during the past few decades^{5,6}. These are not clearly defined terms, but in general, a systems perspective looks at multiple — often numerous — components that interact with each other in potentially complicated ways. Somewhat confusingly, the terms systems immunology and systems biology are also used to describe the analysis of complex data sets, such as high-dimensional -omics data using phenomenological models.

Mechanistic simulation models explicitly specify processes describing the mechanisms of interaction between system components. Usually, these models are highly simplified — but done well, still very powerful — abstractions of the system under study. Engineering and the hard sciences provide some of the best examples for

this approach. For instance, equations describing electric circuits can be simulated to study and predict the behaviour of an actual electronic component. The advantage of this kind of model is that they can provide mechanistic insights, leading to a better and deeper understanding of the system, to a point where the model might allow for very precise predictions. The main disadvantage of this approach is that model construction requires considerable knowledge (or at least assumptions) about the system and how its components interact.

Both phenomenological and mechanistic models are useful tools with distinct advantages and disadvantages. Deciding which type of model to use depends on the question and study system. It is common to start with phenomenological models, to determine patterns and obtain clues regarding the underlying processes and mechanisms, and then to move to a mechanistic model to analyse those processes, their interactions and the resulting outcomes in more detail. In this Review, we will focus on mechanistic simulation models. For the purposes of this article, we will simply use the term ‘model’ to mean a model that describes the dynamics of the components of a system in an explicit and mechanistic way through mathematical equations or computational algorithms. Those models are generally studied by simulating them on a computer. There are different ways that such models can be implemented. The most common types used in immunology are compartmental models, in which each compartment tracks the size of a given biological entity of interest — for example, pathogen load or cytokine concentrations. The most common way to implement a compartmental model is with ordinary differential equations. We will therefore focus on those. BOX 1 introduces compartmental models, while BOX 2 briefly describes alternative modelling approaches.

Uses for models

There are different ways one can categorize the uses of mechanistic simulation models. FIGURE 1 shows one way of conceptualizing different model uses, namely for exploration (also called hypothesis generation), fitting (also called statistical inference) and prediction (also called forecasting). Often, a single project might involve more than one use of a model. We describe these different model uses below and then explain briefly how they are often used iteratively. To make things more tangible, we first introduce and describe two very simple models, one for viral (and other intracellular) infections, and one for extracellular bacterial infections. We then use these example models to illustrate different modelling tasks. We keep the two models purposefully very simple and generic; thus, they should not be considered as representing a detailed model for a specific pathogen (though, as we point out, models as simple as these have been successfully used to answer scientific questions of interest).

A simple model for viral infections

For the viral infection model, we track the numbers of infected and uninfected cells and the levels of free virus. Other details, such as any aspects of the immune response, are ignored in this simple construction.

Box 1 | Compartmental models

Compartmental models are the most widely used and simplest type of simulation model. Such models track the total amount of entities (for example, pathogens or cells). Each entity that one wants to track is assigned a variable and generally is given a mathematical equation that describes how the variable changes. Consider a very simple model in which we track a population of a single entity, P . We model two processes, birth and death (see the figure). A possible equation describing the dynamics of P is

$$P_{t+dt} = P_t + dt(gP_t - dP_t).$$

In words, the population P_{t+dt} at some time step in the future, dt , depends on the number P_t at the current time, t , and the net of their growth (at rate g) and death (at rate d) in the time interval dt . Once we specify growth and death rates, the size of P at some starting time (usually $t=0$) and a time step, we can compute, either manually or with a computer, the value for P at any future time. For instance, if we started with $P=100$ at time $t=0$, with growth and death rates of $g=4$ and $d=2$ per day, and took time steps of $dt=1$ day, we would have 300 at day 1, 900 at day 2 and so forth.

This equation is called a discrete-time model, since we move forward in time in discrete time steps dt . A common alternative to the discrete-time model is a continuous-time model, formulated as an ordinary differential equation (ODE). We can move from the discrete-time model to a continuous-time model by rewriting the equation as

$$\frac{P_{t+dt} - P_t}{dt} = gP_t - dP_t.$$

If we now let the time step become infinitesimally small, we arrive at an ordinary differential equation

$$\frac{dP(t)}{dt} = gP(t) - dP(t).$$

It is common not to show the explicit time dependence of the variable, and to replace the d/dt term with a dot, leading to

$$\dot{P} = gP - dP,$$

which is the notation we use here. Each quantity that one wants to track is assigned an equation. The left side indicates the instantaneous change of the specified variable, and the right side indicates the processes that lead to change. For small time steps, the discrete-time and continuous equations lead to the same results. In fact, when simulating an ODE on a computer, the underlying algorithm employs a smart and efficient version of a discrete-time model.

In general, we want to track more than one entity. Each entity whose change we want to track over time is assigned a variable (a compartment) and given a differential equation. The two example models shown in FIG. 2 and FIG. 3 are three- and two-compartment models.



Box 2 | Beyond ordinary differential equation models

Models based on ordinary differential equations (ODE) are the most common types of models used to study the dynamics of infections and immune responses. These models are easy to implement and fast to simulate, and it is fairly straightforward to fit them to time-series and count data. However, these models have several inherent limitations.

First, ODE models are deterministic, which means they do not account for the inherent randomness present in essentially all biological systems. If you have reason to assume that, for a given system under study, this randomness plays an important role, you need to use a stochastic model. Stochasticity becomes especially important at low numbers. As such, stochastic models become especially important if you want to study emergence or extinction dynamics. The disadvantage of stochastic models is that they are harder to analyse mathematically and harder to fit to data, despite the recent emergence of software that helps make fitting such models simpler (for example, the R package POMP)⁵³. Since stochastic models require multiple simulations to obtain the distribution of outcomes (versus a single simulation run for a

deterministic model), they also take longer to run. Related to this issue, ODE models treat variables as continuous and allow values that are below 1 (such as a fraction of a pathogen). The right kind of stochastic models treat variables as discrete units, where a drop from say one pathogen would result in no pathogens, and thus the end of an infection.

Second, ODE models are in a sense spaceless and assume homogeneous mixing of the different entities/variables that are tracked. If you want to include some notion of space (for example, to simulate multiple sites in an infected host or have a detailed 2D or 3D model of a specific site), you might need to change your model type. A simple extension that still used ODEs would allow for the inclusion of some notion of space in a limited form, by formulating sets of ODEs that described different sites and including terms that allowed for the migration of entities (cells, pathogens) between sites. This has the advantage that you can still use ODE models, at the cost of having space represented in an approximate manner. A more accurate representation of space could be obtained by implementing partial differential equations (PDEs), which, in addition to tracking changes in the variables over time, would also track changes with respect to one or several spatial directions. Unfortunately, PDEs can be tricky to work with, and the way you can account for spatial features is often not well suited for immunological questions. Agent-based models (ABMs) (discussed next) are a good and flexible approach for explicitly modelling spatial features in immunology.

Third, ODE models are compartmental models: they only track the total number of entities (pathogen, cells), not individuals. If you want to track individual entities explicitly, you need to switch to ABMs (also called individual-based models). In such models, some or all entities are tracked individually. Their diverse individual behaviour is determined by parameters usually sampled from probability distributions, allowing for variations among individuals. At the same time, these entities interact in some defined space, thus making ABMs a good choice if you want to track dynamics in a given spatial geometry explicitly. The figure shows a viral infection ABM, in which uninfected cells (green) are placed on a 2D grid (for example, an area of the epithelium). Virus (blue) diffuses on this grid and can infect cells. The infected cells (red) produce new virus. The model also includes T cells (white), which can kill infected cells (see REF⁵⁴ for more of the model details).

ABMs allow for great potential detail and realism in the model. However, compared to ODE models, ABMs are more difficult to write and harder to analyse, and they take longer to run. Furthermore, because these models tend to be more detailed and complex, they usually contain many parameters. Each parameter needs to be given a value based on the system you want to simulate. This means you need to have a lot of quantitative information about a given system before you can build an accurate ABM. Additionally, fitting ABMs to data is substantially more complex than fitting compartmental systems.

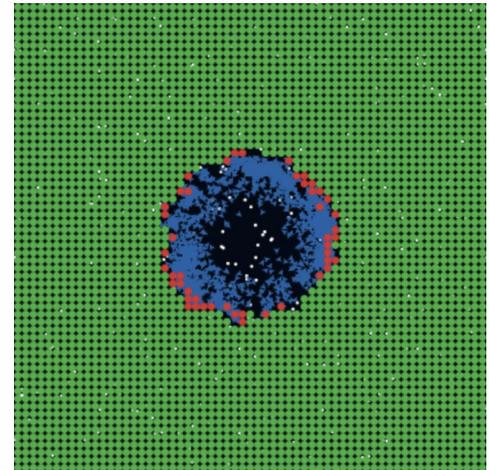


FIGURE 2a shows a graphic representation of such a model. The model shown in FIG. 2a can be translated into equations and computer code in several different ways. The most commonly used implementation is through a set of ordinary differential equations (ODEs). The equations for this model are shown in FIG. 2b. Once the model has been implemented on a computer and the values for the model parameters and variable starting conditions are specified, we can simulate the system. FIGURE 2c shows examples of different dynamics that can be obtained using this model. Despite its simplicity, this type of model has been successfully applied to the study of chronic infections, such as HIV and hepatitis C virus (HCV)^{7,8}, and acute infections, such as influenza virus^{9–13}. We discuss a few of these studies in some detail below.

Model for extracellular bacterial infections

For the bacterial infection model, we track bacteria and the immune response. The latter is implemented in an abstract manner; you can consider it to represent the total immune response or a specific component of importance for a given system. The model is a type of predator–prey model in which the bacteria are the prey and the immune response is the predator. Such models have been widely used in ecology¹⁴. FIGURE 3a shows a graphic representation of such a model, and FIG. 3b shows the set of ODEs corresponding to this model. FIGURE 3c shows two model simulations using different parameter values. The step oscillations seen in bacterial load and immune response for the first scenario are biologically unrealistic, as is the fact that the number of bacteria drops below 1 (a feature of ODE models that is important to note and potentially

Systems thinking and systems modelling

The concept that for systems with many components interacting in complex ways, studying only a small part of the system does not provide a full understanding of the overall system dynamics, and thus that to fully understand the system, one needs to study it in its entirety, often with the help of models.

Systems immunology and systems biology

The application of systems thinking and modelling to immunology and biology.

Compartmental models

A type of model in which one only tracks total numbers for the entities of interest — for example, total number of bacteria and immune response. This contrasts with agent-/individual-based models (BOX 2).

Partial differential equations (PDEs)

A type of differential equation that tracks changes with respect to at least two directions; if used in immunology, these directions are generally time and space in one, two or three dimensions.

Agent-based models (ABMs)

Models in which each entity of interest is tracked and simulated individually, instead of tracking total numbers only.

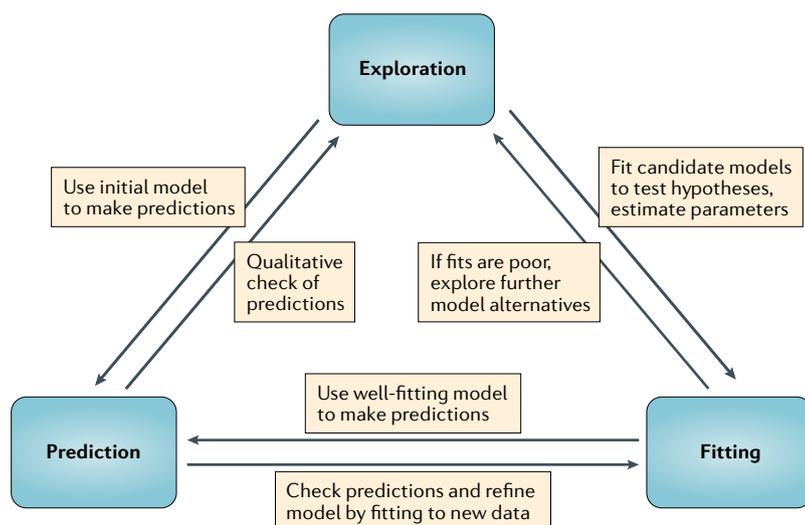


Fig. 1 | **Uses for mechanistic models.** A schematic illustrating a way to categorize and conceptualize different uses of mechanistic models.

to address; see BOX 2). The second scenario, with different parameter values, is more likely to capture the basic dynamics for a real infection process. It is sometimes possible that a model does not produce outcomes that are consistent with the data for any choice of biologically reasonable model parameter values and starting conditions. This means that you have not yet captured all the important mechanisms and processes governing the system under study, and thus the model needs further refinement. We discuss this further below. Models like this have been applied to study of the dynamics of *Mycobacterium tuberculosis*¹⁵ and malaria¹⁶ infections, the interactions between cancers and the immune response¹⁷, and the impact of drugs on *Staphylococcus aureus* infections¹⁸. We discuss a few of these studies in more detail below.

Using models for exploration

If one knows enough about a system to postulate specific processes and mechanisms but does not have a good understanding of how the interactions between different processes affect the overall outcomes, building and analysing a model can be a useful approach. Using models in this way provides a good way to gain some intuition of how a system functions and to generate new hypotheses. It is advisable to keep models simple initially and to increase model complexity as your understanding of the system increases.

As an example of this approach, we consider the extracellular bacterial infection model introduced above and explore how a change in the rate at which the immune response increases in response to bacterial load (parameter r of the model) influences the peak bacterial burden (maximum of variable B). To do so, we implement the model on a computer and then simulate it for different values of r , with all other parameters and initial conditions kept fixed. For each simulation, we obtain a time series for bacteria and immune response like the one shown in FIG. 3c. From this time series, we record the peak bacterial load for each value of r . The results are shown in FIG. 4a.

Such an analysis allows you to explore how different components of the system interact to influence the outcomes. Because this is such a simple model, you might have expected a decrease in peak bacterial load as immune activation increases, even without performing the simulations. However, it might have been difficult to predict the shape and magnitude of the relation. Even basic intuitions can be harder to form as the systems under study, and the models representing them, become more complex.

Using models to fit data

Although models should be built on the basis of the best biological information available, once a model is built and parameter values are determined, it can be analysed without the need for data. To assess the quality of the model, one needs to compare its results with known biological data. This can initially happen qualitatively, which is a common approach during the exploratory stage.

Going beyond qualitative comparisons requires fitting models to data, and thus performing rigorous statistical inference. All the statistical machinery available for phenomenological models can be used to fit mechanistic models. The difference between fitting phenomenological and mechanistic models to data is that the latter allow for more direct testing and possibly for the rejection of specific postulated mechanisms. This provides a potentially deeper understanding of the system than would be possible with phenomenological models alone.

To demonstrate fitting, we consider viral load data from groups of individuals infected with influenza who receive neuraminidase inhibitors early, late or never (symbols in FIG. 4b, for which data were extracted from REF.¹⁹). Assume that you want to investigate potential mechanisms of drug action and postulate the following two hypothetical mechanisms: one, the drug prevents virus entry and new infection of cells; two, the drug reduces the rate at which infected cells produce new virions. To investigate these two mechanisms, you can build two alternative models by incorporating two parameters into the model described above for viral replication. Those parameters, e_1 and e_2 , correspond to the drug's mechanism of action, represented by hypotheses 1 and 2, respectively (see the FIG. 4b caption for details).

Instead of fixing the model parameter values — as is done during model exploration — (some of) the parameters are allowed to vary and are determined by the fitting routine. By fitting a model with parameter e_1 present and another with parameter e_2 present, you can test which mechanism (as implemented in your model) describes the data better. Of course, it could be that the data are best described with both mechanisms e_1 and e_2 acting, as they are not mutually exclusive, or that even with both mechanisms present, the model fails to properly describe the data. For simplicity, we do not test those model variants here.

FIGURE 4b shows the best fit of the model for either mechanism 1 or 2. The two versions of the model are fit to each virus load time series, with either e_1 or e_2 turned on at the indicated times (29 hours post-infection for early treatment, 50 hours post-infection for late treatment). The figure shows that models with either mechanism

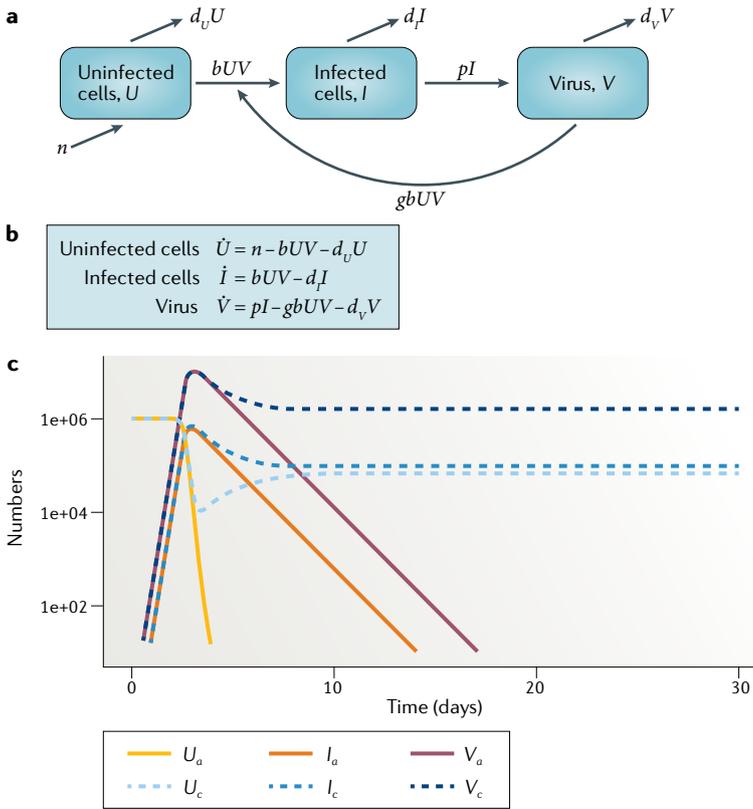


Fig. 2 | A simple model for viral infections. **a** | Diagram for a simple viral infection model. The model tracks changes over time for three quantities (the model variables/ compartments); namely uninfected cells, infected cells and free virus. This model allows for the following processes to occur: uninfected cells are produced at rate n and die naturally at rate $d_v U$. Uninfected cells become infected by virus at rate b , which leads to the removal of a virion and an uninfected cell and the creation of an infected cell (the model term bUV). The extra g in front of this term in the virus equation allows one to account for different experimental units (plaque-forming units or similar measures for virus, versus number of cells) and to ensure that all units in the equations are consistent. Infected cells produce virus at rate p and die at rate d_i . Free virus can infect new, uninfected cells or is cleared by the immune response and other mechanisms (for example, mechanical transport) at rate $d_v V$. **b** | Ordinary differential equation implementation for the simple virus infection model. Each equation tracks the change of a variable (compartment) over time. The left side denotes change of the specified variable with time. The right side describes the processes (also called flows) that lead to change. Positive inflows lead to increases in the variable, whereas negative outflows lead to decreases. Each term on the right side represents a specific biological mechanism — often in an abstract manner. **c** | Acute and chronic infection dynamics of the simple viral infection model. The variables labelled U_a, I_a and V_a show the uninfected, infected and virus compartments for an acute infection. For an acute infection, one generally assumes that over the timescale of the infection, the natural death of cells and, most importantly, the creation of new uninfected cells is negligible, leading to $n = d_v U = 0$ in the model. This model can produce acute infection dynamics, but it does not allow for chronic infections. If one allows the ongoing creation (and, to prevent runaway growth, also background death) of cells, the model can produce a chronic infection scenario. This is shown by the variables labelled U_c, I_c and V_c . For the chronic infection scenario, we set the rate of production of new uninfected cells to $n = 10^5$ per day and the rate of natural death to $d_v U = 0.1$ per day (the inverse of which corresponds to the average lifespan of the cells, here 10 days; the half-life sometimes alternatively used contains an extra factor of $\log(2)$). All other values are the same for the two scenarios. The initial conditions for the variables are $U_0 = 10^6$ uninfected cells, $I_0 = 0$ infected cells and $V_0 = 1$ infectious virion. The parameter values are the virus production rate, $p = 100$ per infected cell per day; the death rate of infected cells, $d_i = 1$ per day (corresponding to a 24-hour average lifespan); a virus clearance rate, of $d_v V = 6$ per day (virus lifespan of 4 hours); rate of infection of $b = 10^{-6}$ per day; and conversion factor $g = 1$, which assumes that the model tracks virus in units of infectious virions. The model is simulated for 30 days.

present come reasonably close to the data. You can discriminate between the model fits with different statistical approaches. A frequently used measure is Akaike’s information criterion (AIC), which tries to quantify the quality of a model fit, with lower values indicating a better-fitting model among a candidate set of models fit to data²⁰. Computing the AICs corrected (AICc) for small sample size for the two models gives $AICc_1 = -56$ and $AICc_2 = -82$. Thus, on the basis of AIC, the model fitting suggests that mechanism e_2 — that is, drug-induced reduction of virus production by infected cells — fits the data better. It is important to note that when doing model comparisons, the (statistically) ‘best’ model is not necessarily a ‘good’ model. Determination of the overall quality of the model cannot be done purely on statistical grounds, but requires a scientific, not a purely statistical, judgement call to decide whether a model can be considered a reasonable approximation of the underlying system.

In addition to discriminating between different models, and thus mechanisms, fitting provides estimates for the model parameters. In contrast to most parameters in phenomenological models (for example, regression coefficients), the parameters in mechanistic models often have direct biological meaning. Here, you find that for the more likely mechanism (mechanism 2), the value measuring the strength of the drug is $e_2 = 0.98$, suggesting that the drug is highly effective and reduces virus production by 98%. Of course, an important caveat for the interpretation of the estimated parameters is that the model needs to provide a reasonable approximation of the real system.

Despite rapid improvements in the capability and user-friendliness of software, fitting mechanistic models to data can still be technically challenging. It also requires a good match between available data, the model and the scientific question. On the plus side, a fitting approach allows for direct testing of different hypotheses, formulated as different model mechanisms or model variants, and provides estimates for potentially important biological quantities.

Using models to make predictions

Through some combination of the model use approaches just described, you might be able to understand a system well enough that you can build a model that provides a fairly accurate approximation of the real system. You can then use the model to perform *in silico* experiments: you can make predictions about what might happen to the system if some components were altered, such as through the introduction of a drug. Using models for prediction or forecasting follows essentially the same approach as exploratory model use. The difference is that predictive modelling requires confidence that the model can decently approximate the real system.

Assume that the simple bacteria model accurately captures the dynamics of some real system of interest. In that case, the result shown in FIG. 4a could be interpreted as making predictions of how a change in immune activation rate (possibly mediated by some drug) influences the peak bacterial burden.

Often, if models are used for predictions, it is useful to provide estimates of uncertainty in the predictions.

In silico experiments

Using models to explore and predict outcomes for scenarios for which there are currently no data — for example, the introduction of a drug into the system described by the model.

A common approach for obtaining such estimates is through an uncertainty and sensitivity analysis^{21,22}. For ODE models like the ones described here, such an analysis involves varying the model parameters in ranges that are considered biologically reasonable. For each set of model parameters, you run the model and compute the quantities of interest. This provides a set of model predictions, one for each set of model parameters. This distribution of model results provides some measure of

uncertainty in the model predictions, due to uncertainty in the underlying model parameters.

Compared to real experiments, these in silico approaches are much faster and cheaper, and generally they have no ethical implications. The major caveat with predictions obtained from such models is that they are only reliable insofar as the model properly captures the important features of the real system. Thus, an iterative process is usually employed, of model predictions, comparisons to data and further exploration and refinement of the models. This can lead to increasingly better and more predictive models. Weather forecasting is a domain where this approach has proven to be very successful.

Model use in practice

Although it is possible to use a model in only one of the ways just described, iterative use as illustrated in FIG. 1 is common. Typically, one or several simple models are initially built to explore the dynamics of the system of interest and are qualitatively compared to what is known about the biological system. If no suitable data for fitting are available, the model can be used to generate hypotheses and make predictions, which then should be tested by comparing to data, either qualitatively or through statistically rigorous fitting. If suitable data are available, the exploration stage is often followed by fitting the most promising candidate model(s) to the data to obtain statistical support for specific models, as well as estimates for the model parameters. These statistically supported models can then be used to make predictions about as-yet-unobserved behaviour of the system (for example, simulating the removal of a cytokine in the model). Predictions should then be tested with further data, likely leading to further model refinements.

A few real-world examples

Mechanistic simulation models of the type we just described are increasingly used in immunological and infectious disease research. The literature is too vast to fully review it. We therefore provide a review and brief summary of a few prominent examples and pointers towards further discussions of such models.

Nowak and colleagues used a simple model similar to the viral infection model described above to gain general insights into HIV infection dynamics²³. A finding from their analysis was the suggestion that viral sequence diversity is positively correlated with virus load. In another study, model fitting was used to allow the authors to estimate the rates and total amount of HIV virus production, as well as CD4⁺ T cell numbers and virus half-lives²⁴. These findings helped confirm the futility of single-drug treatment. Another study considered the impact of interferon (IFN) for HCV-infected patients; by fitting a simple model, the researchers determined that IFN's main mode of action is to block virion production²⁵. That study also provided quantitative estimates for the efficacy of IFN treatment as well as virus half-life. For more examples and further details on such models applied to HIV and HCV, see, for example, REFS^{7,8}.

The same types of viral infection models have been used to study influenza. In REF.²⁶ the authors used a combination of models and data to show that the fitness of

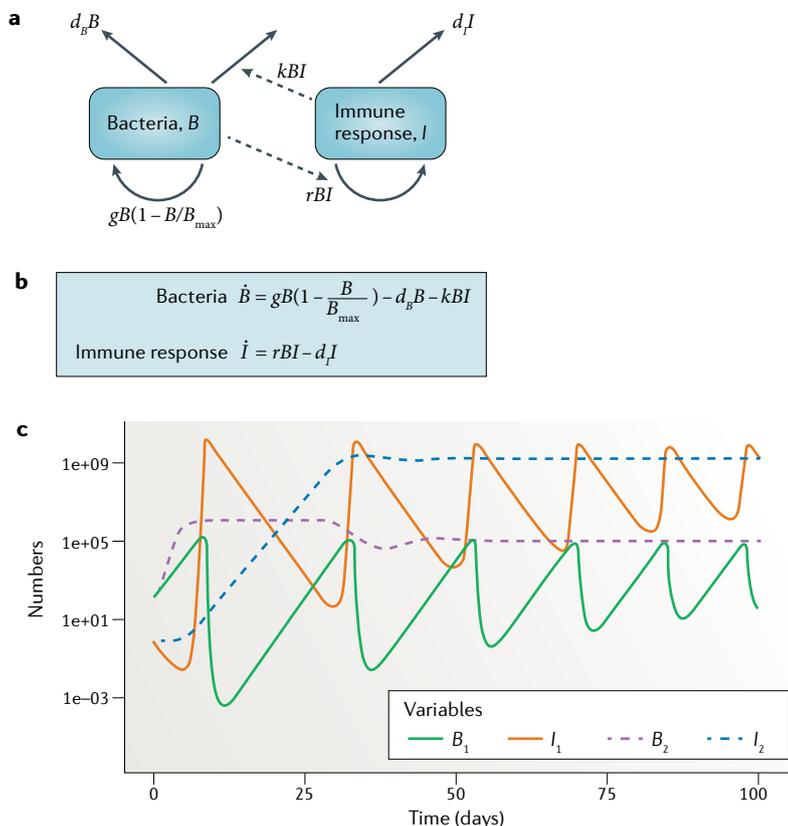


Fig. 3 | A simple model for extracellular bacterial infections. a | Diagram for the simple bacterial infection model. The included compartments, whose numbers are tracked over time, are bacteria, B, and the immune response, I. The following processes are implemented: Bacteria grow at a maximum rate *g*, and growth slows down to zero as they approach the value *B*_{max}. The bacteria also die at some fixed rate *d*_b and are additionally cleared by the immune response at rate *k*. The immune response is modelled in a very abstract way with a single equation. It can be interpreted as a specific biological component that is dominant for a given system, or could be considered as the total strength of the response. The immune response has a per-capita growth proportional to the presence of bacteria at rate *r*, and a decay rate of *d*_i. **b** | Ordinary differential equations for the simple bacterial infection model. The compartments being tracked are the total number of bacteria and the total strength of the immune response and their changes over time. The right side describes the processes (also called flows) that lead to change. Positive inflows lead to increases in the variable, and negative outflows lead to decreases. Each term on the right side represents a specific biological mechanism — often in an abstract manner. See the model diagram for a description of the different processes. **c** | Dynamics of the simple bacteria model for two different scenarios. In both scenarios, the initial values for bacteria and immune response are *B* = 100 and *I* = 1, and the model is run for 100 days; we assume that the maximum rate of bacteria growth is *g* = 2 per day (cells divide every 12 hours); the value at which growth will stop is *B*_{max} = 10⁶; and the rate at which the immune response kills bacteria is *k* = 10⁻⁹. The remaining parameters differ between the two scenarios. For scenario 1 (lines *B*₁ and *I*₁), bacteria die at a rate of *d*_b = 1 per day (average lifespan of 1 day), and the immune response decays at a rate of *d*_i = 1 per day and grows at a rate of *r* = 10⁻⁴ per day. For scenario 2 (lines *B*₂ and *I*₂), those values are *d*_b = 0.1, *d*_i = 0.1 and *r* = 10⁻⁶.

neuraminidase-resistant H1N1 2009 pandemic influenza strains was similar to that of drug-susceptible virus strains. Such models helped shed light on the dynamics of co-infection with influenza virus and *Streptococcus pneumoniae*. Analysis of a co-infection model suggested that around four to six days after influenza virus infection, secondary bacterial infection can occur at a much lower exposure dose²⁷. Another study²⁸ discriminated between different mechanisms of pathogen interaction by fitting a model to data and found that increased virus

release was likely in the presence of bacteria. Further examples and details of influenza models can be found in REFS^{9–13}.

Models like the bacteria model discussed above have been used to study malaria and tuberculosis infections. In one such study²⁹, the authors used a simple discrete-time model, fitted to data, to elucidate the relative roles of target-cell limitation versus immune responses during malaria infections. A four-equation differential equation model³⁰ allowed the prediction of how changes in asexual parasite densities influence gametocyte conversion rates. In a study of tuberculosis³¹, the authors used a detailed model to explore and predict the impact of depleting key cytokines on the infection dynamics. They found, for instance, that depletion of IL-10 and IL-4 during latency led to a large increase in bacterial density. Further examples of simulation models applied to malaria and tuberculosis can be found in REFS^{32,33}.

Models are also often used to study the effects of antimicrobial drugs. This approach is generally referred to as pharmacokinetic–pharmacodynamic (PK–PD) modelling³⁴. While most PK–PD models include equations for the processes describing the pathogen and the drug dynamics, only a few also model components of the immune response. In an example of a model that includes PK–PD and the immune response³⁵, use of a model suggested that high-dose, extended treatment with antibiotics is often the best strategy for reducing resistance emergence. For additional examples of models that include components of the immune response, see REFS^{36–38}.

Another area in which such models have been used is in probing the dynamics of T cells. As an example, stochastic simulation models were used in REF.³⁹ to determine that random differentiation and division events drive CD8⁺ T cell diversification. Application of another model helped characterize divergent subpopulations of CD4⁺ effector and central memory T cells⁴⁰. REFS^{41,42} provide further examples.

These types of models have also been successfully applied to the study of non-infectious systems. A compartmental simulation model applied to chronic myeloid leukaemia showed that the drug imatinib inhibits the production of new leukaemic cells but does not deplete existing cells⁴³. In REF.⁴⁴, the authors combined models and data to determine the existence of distinct subpopulations of haematopoietic stem cells with different turnover rates.

These examples are just a few among many. For further examples of such models applied to specific systems or questions, see, for example, the recent collections of articles introduced in REFS^{45,46}.

How to apply models

Reading thus far might have persuaded you that the kinds of models we describe could be useful for your research. Here are some suggestions for the next steps.

Form a team. We described what simulation models can and cannot do. If you have a question that lends itself to this kind of modelling, arguably the best option is to team up with a modelling expert. The number of scientists using mechanistic simulation modelling approaches

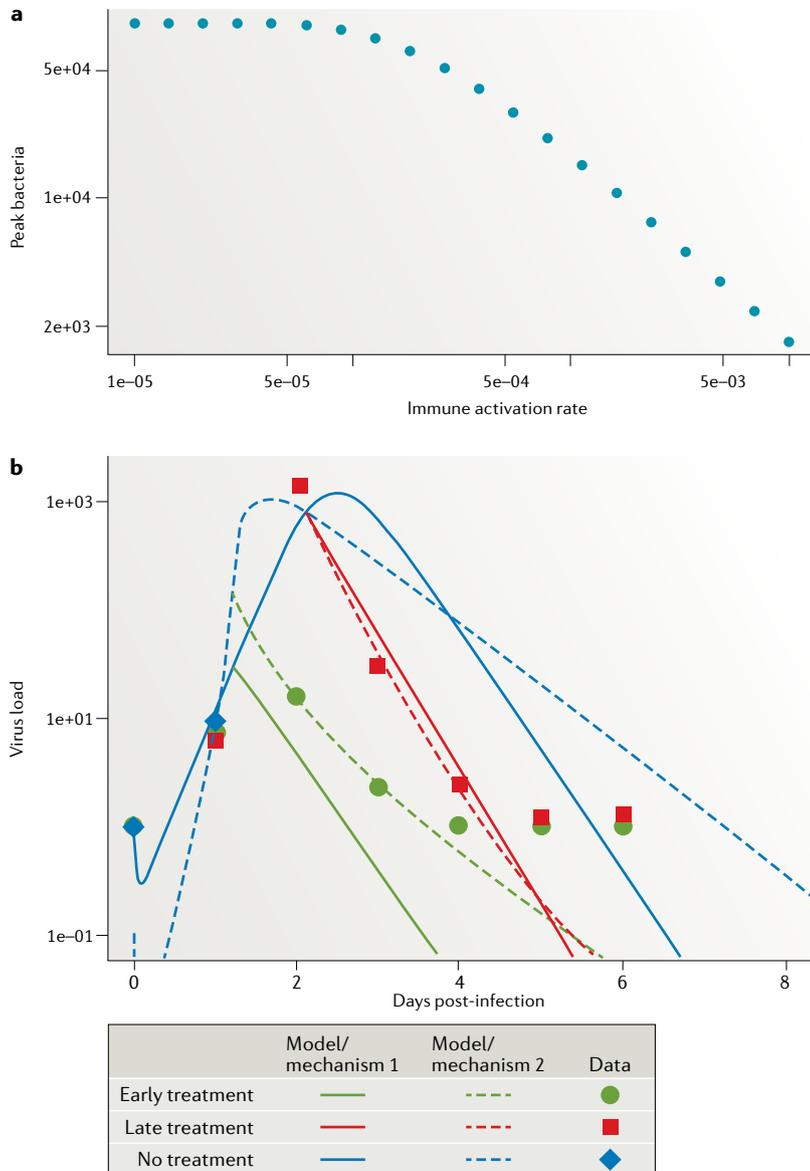


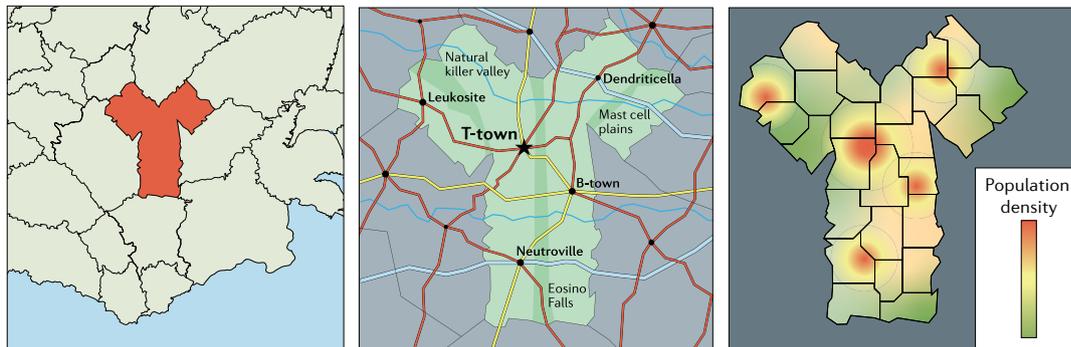
Fig. 4 | **Different uses of mechanistic models.** **a** | Bacteria peak numbers as a function of immune response activation rate (model parameter r). **b** | Influenza virus load data and model fits in the absence and presence of neuraminidase inhibitor treatment. The symbols are data from REF.¹⁹. The model is the acute virus infection model introduced above ($n = d_U = 0$), with the following changes: to implement mechanism 1, we replace the infection terms bUV and $gbUV$ with $b(1 - e_1)UV$ and $gb(1 - e_1)UV$, where e_1 describes the strength of the first mechanism. To implement mechanism 2, we replace pl with $p(1 - e_2)l$, with e_2 describing the strength of the second mechanism. Solid lines show the best fits of the model with mechanism e_1 , and the dashed lines show fits of the model with mechanism e_2 .

Box 3 | How detailed should my model be?

As the immune response is very complex, you might be inclined to build very complicated and detailed models, to be as realistic as possible. Although more detailed models can indeed be more realistic, there are several drawbacks. First, as models get larger, they contain more parameters. Each parameter needs to be given a numeric value to allow you to run simulations. You can try to obtain the parameters from the literature, but this information is often not available. Alternatively, you could fit the model to data and try to estimate the parameters. However, with the kind of data typically available, you can usually only estimate a few parameters with some level of certainty. Furthermore, larger models are harder to implement, take longer to run and are more difficult to analyse. With too many parts present, it can be hard to understand how different components interact with each other and affect the outcomes of interest.

A good analogy for determining the right model is the use of maps. Maps are models of the real world. They serve specific purposes, and it is important that a given map be useful for the intended purpose. Consider the three maps (models) of the fictional country of Antibodia (see the figure). If you want to know where this country is located, the left map is useful. If instead you want to know how to drive from T-town to Dendriticella, the middle map would be the most useful. If you want to know where most people live in this country, the right map is most useful. It is the same 'system' under consideration (the country of Antibodia), but depending on the question, different maps (models) are needed. Analogously, for the same biological system under study (for example, a specific pathogen and host), different types of models that include and exclude different details of the systems are needed, depending on the question you want to answer. The usefulness of maps (and models) is that they capture the information that is needed for a specific situation, while ignoring details that are not important for a given question, thus producing the right level of complexity.

To build models that are suitable to study a particular system, model builders need to be knowledgeable about the system they want to study or to collaborate with subject matter experts. Building a good model needs to follow the Goldilocks principle: If a model is too simple, it likely does not approximate the real system very well. If a model is too complicated, it is hard to build and analyse, and might not lead to much insight (that is, the model is a big black box). The goal is to get the model just right regarding both size and complexity. Unfortunately, no recipe or formula exists specifying how to build such a 'just right' model. Successful model building and analysis is often iterative. After a model has been built and studied, it might become clear — usually by comparing the model with data — that important components or interactions have been ignored or not been included correctly. This leads to model modification and refinement. This back and forth between model and data can happen over multiple iterations.



has been increasing considerably over the past decades. These individuals are distinct from biostatisticians and bioinformaticians, although there is often considerable overlap. Modellers are often ready to apply their skills and tools to interesting immunological questions. To make this team approach successful, each collaborator needs to understand enough about the other person's domain to engage in a meaningful manner. Thus, modelers need to understand enough biology and immunology to build reasonable models, and immunologists need to understand enough about the models to know the models' strengths and limitations and to assess their usefulness for a given situation. We hope this article helps with the latter aspect.

Build your own model. Some readers might want to try building and analysing models themselves. Before doing so, consider whether a mechanistic model is the right tool to address your question. At times, a question may

best be addressed by experiments and collecting more data. At other times, a phenomenological model might be the right approach. If a simulation model is suitable for your question, determine what type of mechanistic model you need. Compartmental models implemented as ODEs (the models we have focused on in this review) are often a good first choice. However, for certain questions and scenarios, those model types are not suitable, and you might need a different type of model. See BOX 2 for a brief discussion.

Next, you need to decide how much detail to include. We recommend starting simple and increasing complexity as needed (BOX 3). As you build your model, be clear about the assumptions you make, and ensure that your assumptions can be justified on the basis of what is known about the biology of the system. If several potential processes may be reasonably assumed, it might be suitable to explore multiple models, each encoding one or some of the biologically reasonable mechanisms

Box 4 | A guide for model building

If you decided that a compartmental model is suitable for your project, and you would like to implement one yourself, here are steps you can take to implement such a model and check its suitability:

1. Determine which compartments and variables to include.
2. Draw a diagram with variables as boxes. Add all processes as arrows.
3. Translate the diagram into a set of differential equations. Each arrow in your diagram should correspond to a flow term in your equations.
4. Go back and forth between your equations, your diagram and your biological description of the system. Make sure everything is consistent.
5. Check that the model behaves reasonably by investigating equations for 'extreme' situations. For instance, if a given variable is zero, no outflow should occur from this compartment. Similarly, you probably do not want a model in which any quantity can grow without bounds. Next, check that nothing biologically unreasonable can occur. For instance, ensure that in the absence of a pathogen (those variables set to 0), the immune response does not grow. You can do some of these checks 'on paper', by setting certain variables to zero and thinking through the model behaviour. Alternatively, you can implement the model first and then test numerically that nothing biologically unreasonable happens. The larger your model is, the more important it is to isolate and test specific parts of it and ensure they behave properly.
6. Obtain estimates for the parameter values and initial conditions for your model from the experimental literature. Contrary to the expectations of many novice modellers, this step is often quite difficult and time-consuming. Direct estimates for the model parameters might not be available. You then need to use the best available experimental evidence, combined with assumptions based on your understanding of the system, to supply values for specific parameters. For parameters whose values are not well known, the uncertainty analysis described above can be a helpful approach later in the modelling process.
7. Implement the model on a computer, and explore. As needed, make further modifications.
8. Depending on your needs and data availability, move on to fitting or predictions.

(see, for instance, the two drug mechanism models described above).

A few specific pointers regarding practical steps towards implementing the models are given in BOX 4. Once you have one or several starter models built, you can explore their behaviour, use them to generate hypotheses and predictions, or fit them to available data for hypothesis testing. During this process, you will likely further refine and improve your model.

While coding is generally required, there are ways to build models that do not require writing computer code. Some software, such as Berkeley Madonna, Stella or more specialized tools⁴⁷, allows you to graphically

build and run models. However, be prepared for the likelihood that at some point you will need to engage in some level of code writing. Before you embark on learning a specific programming language, it is worth considering which one to learn and use. Currently, two of the most popular general-purpose languages for scientific computing are R and Python. Both languages are free and highly flexible, and both have large user and developer communities. Commercial products such as MATLAB or Mathematica are also frequently used.

One of the authors has developed a free software package called Dynamical Systems Approaches to Immune Response Modelling (DSAIRM)⁴⁸. DSAIRM is implemented as an R package. The main goal of DSAIRM is to provide a user-friendly tool to learn about modelling in immunology, without the need to write code. It also provides access to the code for all the models implemented in DSAIRM, which you can then modify for your specific research question. Further details can be found on the package website⁴⁸ and in REF.⁴⁹.

Next steps. To learn more, several books^{8,50–52} and review articles^{7,41,46} provide further details on mechanistic simulation modelling. Beyond additional reading, a great way to learn the material is by actively engaging with it. The above-mentioned DSAIRM R package is meant for that purpose. It is quick and easy to install, and you can use it to explore and learn a range of models and concepts related to infection and immunology modelling using a graphical interface.

Conclusion

The type of mechanistic modelling discussed here is used increasingly in biology, and specifically in immunology, microbiology and related areas. The adoption and progress of modelling have been rapid, driven by data of higher quantity and quality, better and easier-to-use computational tools, and a general drive in immunology towards becoming an increasingly rigorous, quantitative discipline. Together with other quantitative analysis approaches, mechanistic simulation models provide a useful set of tools that allow one to investigate mechanisms of interaction and the resulting dynamics for a specific system.

Published online 5 December 2019

<p>1. Jaqaman, K. & Danuser, G. Linking data to models: data regression. <i>Nat. Rev. Mol. Cell Biol.</i> 7, 813–819 (2006).</p> <p>2. Altmann, D. M. New tools for MHC research from machine learning and predictive algorithms to the tumour immunopeptidome. <i>Immunology</i> 154, 329–330 (2018).</p> <p>3. Ching, T. et al. Opportunities and obstacles for deep learning in biology and medicine. <i>J. R. Soc. Interface</i> 15, 20170387 (2018).</p> <p>4. Vigen, T. Spurious correlations. <i>tylervigen</i> http://tylervigen.com/spurious-correlations (2019).</p> <p>5. Narang, V. et al. Systems immunology: a survey of modeling formalisms, applications and simulation tools. <i>Immunol. Rev.</i> 53, 251–265 (2012).</p> <p>6. Arazi, A., Pendergraft, W. F., Ribeiro, R. M., Perelson, A. S. & Hacohen, N. Human systems immunology: hypothesis-based modeling and unbiased data-driven approaches. <i>Semin. Immunol.</i> 25, 193–200 (2013).</p> <p>7. Perelson, A. S. Modelling viral and immune system dynamics. <i>Nat. Rev. Immunol.</i> 2, 28–36 (2002).</p>	<p>8. Nowak, M. A. & May, R. M. <i>Virus Dynamics: Mathematical Principles of Immunology and Virology</i> (Oxford Univ. Press, 2001).</p> <p>9. Beauchemin, C. A. A. & Handel, A. A review of mathematical models of influenza infections within a host or cell culture: lessons learned and challenges ahead. <i>BMC Public Health</i> 11 (Suppl. 1), S7 (2011).</p> <p>10. Smith, A. M. & Perelson, A. S. Influenza a virus infection kinetics: quantitative data and models. <i>Wiley Interdiscip. Rev. Syst. Biol. Med.</i> 3, 429–445 (2011).</p> <p>11. Murillo, L. N., Murillo, M. S. & Perelson, A. S. Towards multiscale modeling of influenza infection. <i>J. Theor. Biol.</i> 332, 267–290 (2013).</p> <p>12. Smith, A. M. Host–pathogen kinetics during influenza infection and coinfection: insights from predictive modeling. <i>Immunol. Rev.</i> 285, 97–112 (2018).</p> <p>13. Handel, A., Liao, L. E. & Beauchemin, C. A. A. Progress and trends in mathematical modelling of influenza A virus infections. <i>Curr. Opin. Syst. Biol.</i> 12, 30–36 (2018).</p>	<p>14. Otto, S. P. & Day, T. <i>A Biologist's Guide to Mathematical Modeling in Ecology and Evolution</i> (Princeton Univ. Press, 2007).</p> <p>15. Antia, R., Koella, J. C. & Perrot, V. Models of the within-host dynamics of persistent mycobacterial infections. <i>Proc. Biol. Sci.</i> 263, 257–63 (1996).</p> <p>16. Kochin, B. F., Yates, A. J., Roode, J. C. de & Antia, R. On the control of acute rodent malaria infections by innate immunity. <i>PLOS ONE</i> 5, e10444 (2010).</p> <p>17. Wilkie, K. P. A review of mathematical models of cancer–immune interactions in the context of tumor dormancy. <i>Adv. Exp. Med. Biol.</i> 734, 201–234 (2013).</p> <p>18. Chung, P., McNamara, P. J., Campion, J. J. & Evans, M. E. Mechanism-based pharmacodynamic models of fluoroquinolone resistance in <i>Staphylococcus aureus</i>. <i>Antimicrob. Agents Chemother.</i> 50, 2957–2965 (2006).</p> <p>19. Hayden, F. G. et al. Safety and efficacy of the neuraminidase inhibitor GG167 in experimental human influenza. <i>JAMA</i> 275, 295–299 (1996).</p> <p>20. Burnham, K. P. & Anderson, D. R. <i>Model Selection and Multimodel Inference</i> (Springer, 2002).</p>
---	--	--

21. Marino, S., Hogue, I. B., Ray, C. J. & Kirschner, D. E. A methodology for performing global uncertainty and sensitivity analysis in systems biology. *J. Theor. Biol.* **254**, 178–196 (2008).
22. Hoare, A., Regan, D. G. & Wilson, D. P. Sampling and sensitivity analyses tools (SaSAT) for computational modelling. *Theor. Biol. Med. Model.* **5**, 4 (2008).
23. Nowak, M. A. & Bangham, C. R. Population dynamics of immune responses to persistent viruses. *Science* **272**, 74–79 (1996).
24. Perelson, A. S., Neumann, A. U., Markowitz, M., Leonard, J. M. & Ho, D. D. HIV-1 dynamics in vivo: virion clearance rate, infected cell life-span, and viral generation time. *Science* **271**, 1582–1586 (1996).
25. Neumann, A. U. et al. Hepatitis C viral dynamics in vivo and the antiviral efficacy of interferon-alpha therapy. *Science* **282**, 103–107 (1998).
26. Butler, J. et al. Estimating the fitness advantage conferred by permissive neuraminidase mutations in recent oseltamivir-resistant a(H1N1)pdm09 influenza viruses. *PLOS Pathog.* **10**, e1004065 (2014).
27. Shrestha, S. et al. Time and dose-dependent risk of pneumococcal pneumonia following influenza: a model for within-host interaction between influenza and *Streptococcus pneumoniae*. *J. R. Soc. Interface* **10**, 20130233 (2013).
28. Smith, A. M. et al. Kinetics of coinfection with influenza a virus and *Streptococcus pneumoniae*. *PLOS Pathog.* **9**, e1003238 (2013).
29. Metcalf, C. J. E. et al. Partitioning regulatory mechanisms of within-host malaria dynamics using the effective propagation number. *Science* **333**, 984–988 (2011).
30. Schneider, P. et al. Adaptive plasticity in the gametocyte conversion rate of malaria parasites. *PLOS Pathog.* **14**, e1007371 (2018).
31. Wigginton, J. E. & Kirschner, D. A model to predict cell-mediated immune regulatory mechanisms during human infection with mycobacterium tuberculosis. *J. Immunol.* **166**, 1951–1967 (2001).
32. Mideo, N., Day, T. & Read, A. F. Modelling malaria pathogenesis. *Cell Microbiol.* **10**, 1947–1955 (2008).
33. Kirschner, D., Pienaar, E., Marino, S. & Linderman, J. J. A review of computational and mathematical modeling contributions to our understanding of mycobacterium tuberculosis within-host infection and treatment. *Curr. Opin. Syst. Biol.* **3**, 170–185 (2017).
34. Drusano, G. L. Antimicrobial pharmacodynamics: critical interactions of 'bug and drug'. *Nat. Rev. Microbiol.* **2**, 289–300 (2004).
35. Ankomah, P. & Levin, B. R. Exploring the collaboration between antibiotics and the immune response in the treatment of acute, self-limiting infections. *Proc. Natl Acad. Sci. USA* **111**, 8331–8338 (2014).
36. Canini, L., Conway, J. M., Perelson, A. S. & Carrat, F. Impact of different oseltamivir regimens on treating influenza a virus infection and resistance emergence: insights from a modelling study. *PLOS Comput. Biol.* **10**, e1003568 (2014).
37. Handel, A., Margolis, E. & Levin, B. R. Exploring the role of the immune response in preventing antibiotic resistance. *J. Theor. Biol.* **256**, 655–662 (2009).
38. Cjini, E. & Brito, P. H. Integrating antimicrobial therapy with host immunity to fight drug-resistant infections: classical vs. adaptive treatment. *PLOS Comput. Biol.* **12**, e1004857 (2016).
39. Buchholz, V. R. et al. Disparate individual fates compose robust CD8⁺ T cell immunity. *Science* **340**, 630–635 (2013).
40. Gossel, G., Hogan, T., Cownden, D., Seddon, B. & Yates, A. J. Memory CD4 T cell subsets are kinetically heterogeneous and replenished from naive T cells at high levels. *eLife* **6**, e23013 (2017).
41. Antia, R., Ganusov, V. V. & Ahmed, R. The role of models in understanding CD8⁺ T-cell memory. *Nat. Rev. Immunol.* **5**, 101–111 (2005).
42. Gerritsen, B. & Pandit, A. The memory of a killer T cell: models of CD8⁺ T cell differentiation. *Immunol. Cell Biol.* **94**, 236–241 (2016).
43. Michor, F. et al. Dynamics of chronic myeloid leukaemia. *Nature* **435**, 1267–1270 (2005).
44. Takizawa, H., Regoes, R. R., Boddupalli, C. S., Bonhoeffer, S. & Manz, M. G. Dynamic variation in cycling of hematopoietic stem cells in steady state and inflammation. *J. Exp. Med.* **208**, 273–284 (2011).
45. Kirschner, D. & Mehr, R. Editorial overview. *Curr. Opin. Syst. Biol.* **12**, iv–vi (2018).
46. Perelson, A. S. & Ribeiro, R. M. Introduction to modeling viral infections and immunity. *Immunol. Rev.* **285**, 5–8 (2018).
47. Germain, R. N., Meier-Schellersheim, M., Nita-Lazar, A. & Fraser, I. D. C. Systems biology in immunology: a computational modeling perspective. *Annu. Rev. Immunol.* **29**, 527–585 (2011).
48. Handel, A. et al. Software DSAIRM: dynamical systems approach to immune response modeling. *GitHub* <https://ahgroup.github.io/DSAIRM/> (2019).
49. Handel, A. A software tool to teach mechanistic modeling to immunologists. *BMC Immunol.* <https://doi.org/10.1186/s12865-019-0321-0> (2019).
50. Wodarz, D. *Killer Cell Dynamics Mathematical and Computational Approaches to Immunology* (Springer, 2007).
51. Bassaganya-Riera, J. *Computational Immunology: Models and Tools* (Academic Press, 2015).
52. Hernandez-Vargas, E. A. *Modeling and Control of Infectious Diseases in the Host* (Academic Press, 2019).
53. King, A. A. et al. POMP: statistical inference for partially observed Markov processes. R package, version 2.3. (2019).
54. Handel, A., Yates, A., Pilyugin, S. S. & Antia, R. Sharing the burden: antigen transport and firebreaks in immune responses. *J. R. Soc. Interface* **6**, 447–454 (2009).

Acknowledgements

We thank A. Souquette and R. Mettelman for comments on an early draft. We also thank the reviewers for their useful feedback. A.H. and P.T. were partially supported by NIH U19AI117891. N.L.L.G. was supported by a National Health and Medical Research Council Program grant (APP1071916) and an Australian Research Council (ARC) Future Fellowship (FT170100174).

Author contributions

A.H., N.L.L.G. and P.G.T. contributed to the writing, review and editing of the manuscript. A.H. and P.G.T. substantially contributed to the discussion of content. A.H. was involved in researching data for the article.

Competing interests

The authors declare no competing interests.

Publisher's note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Reviewer information

Nature Reviews Immunology thanks R. de Boer and R. Regoes for their contribution to the peer review of this work.

© Springer Nature Limited 2019