INTERFACE

rsif.royalsocietypublishing.org

Research

Cite this article: Read MN, Alden K, Rose LM, Timmis J. 2016 Automated multi-objective calibration of biological agent-based simulations. *J. R. Soc. Interface* **13**: 20160543. http://dx.doi.org/10.1098/rsif.2016.0543

Received: 7 July 2016
Accepted: 22 August 2016

Subject Category:
Life Sciences–Mathematics interface

Subject Areas:
computational biology, bioinformatics

Keywords:
computational biology, agent-based simulation, calibration, multi-objective optimization, experimental autoimmune encephalomyelitis

Author for correspondence:
Mark N. Read
e-mail: mark.read@sydney.edu.au

Electronic supplementary material is available online at doi:10.6084/m9.figshare.c.3461745.

THE ROYAL SOCIETY
PUBLISHING

Automated multi-objective calibration of biological agent-based simulations

Mark N. Read[1,2], Kieran Alden[3], Louis M. Rose[4] and Jon Timmis[3]

[1]School of Life and Environmental Sciences, and [2]Charles Perkins Centre, The University of Sydney, Camperdown, New South Wales, Australia
[3]Department of Electronics, and [4]Department of Computer Science, University of York, York, UK

MNR, 0000-0002-1481-4780

Computational agent-based simulation (ABS) is increasingly used to complement laboratory techniques in advancing our understanding of biological systems. Calibration, the identification of parameter values that align simulation with biological behaviours, becomes challenging as increasingly complex biological domains are simulated. Complex domains cannot be characterized by single metrics alone, rendering simulation calibration a fundamentally multi-metric optimization problem that typical calibration techniques cannot handle. Yet calibration is an essential activity in simulation-based science; the baseline calibration forms a control for subsequent experimentation and hence is fundamental in the interpretation of results. Here, we develop and showcase a method, built around multi-objective optimization, for calibrating ABSs against complex target behaviours requiring several metrics (termed *objectives*) to characterize. Multi-objective calibration (MOC) delivers those sets of parameter values representing optimal trade-offs in simulation performance against each metric, in the form of a Pareto front. We use MOC to calibrate a well-understood immunological simulation against both established *a priori* and previously unestablished target behaviours. Furthermore, we show that simulation-borne conclusions are broadly, but not entirely, robust to adopting baseline parameter values from different extremes of the Pareto front, highlighting the importance of MOC's identification of numerous calibration solutions. We devise a method for detecting overfitting in a multi-objective context, not previously possible, used to save computational effort by terminating MOC when no improved solutions will be found. MOC can significantly impact biological simulation, adding rigour to and speeding up an otherwise time-consuming calibration process and highlighting inappropriate biological capture by simulations that cannot be well calibrated. As such, it produces more accurate simulations that generate more informative biological predictions.

1. Introduction

Computational modelling and simulation has emerged as a tool for investigating a wide range of biological systems, spanning immunology [1,2], drug and intervention design [3,4], developmental biology [5] and ecology [6]. Biological simulation is particularly insightful when used to complement traditional methods, such as wet-lab *in vivo* and *in vitro* work; laboratory work generates experimental data and suggests hypotheses that can be evaluated by way of their integration with simulation, which in turn can suggest further experiments or highlight areas of lacking knowledge [7,8]. Well-designed, biologically accurate simulations provide detailed spatio-temporal insight, facilitating observations and assays not possible in the real system; simulation experiments are unhampered by the ethical, practical and financial considerations inherent in biological experimentation. Research programmes integrating wet-lab and simulation methods can offer a greater return on animal experimentation by generating additional insight, and hence easing the burden on experimental animals, in line with the '3Rs' principles (replacement, reduction and refinement).

The agent-based simulation (ABS) paradigm permits detailed and nuanced simulation of biological systems [3,9]. Simulation components are represented as explicit individual entities, *agents*, with unique states that exist within a spatial environment. Rules specifying agent dynamics and the consequences of interaction are provided, and simulation execution allows the system-level consequences of agent-level manipulations to be observed. ABS incorporates stochastic events, and therein reflects the heterogeneity of real-world natural systems. There is scope for specifying very detailed interactions using ABS, at the expense of generating large numbers of parameters: 50+ is not uncommon.

Drawing biologically meaningful conclusions from simulation requires that the mapping of the simulation to the biology is known. This can prove problematic for two reasons. First, simulations are abstract representations of their corresponding real-world systems. For example, there exist at least 19 varieties of T cell, a vital component of the immune system [10]. However, rather than fully capture all their nuanced differences, a simulation is more likely to represent an abstracted subset thereof. As such, experimental measurements on a real-world T cell cannot be assumed to translate directly to its simulation counterpart. Second, complex biological systems are the subject of simulation precisely because they are incompletely understood, meaning that the real-world data supporting simulation design decisions and corresponding parameter values may not exist. Calibration is a critical activity in establishing the link between simulation and biology; parameter values that align simulation and real-world dynamics are identified. Furthermore, an inability to provide a good alignment points to simulation design that does not appropriately capture the biology. Calibration is used to establish a baseline simulation dynamic used as a control in subsequent experimentation, and finding appropriate values is important. Different parameter values will yield different simulation dynamics, and as such influence the conclusions drawn from experiments.

A number of approaches to calibration exist, including manual calibration [11], evolutionary algorithms [12,13], maximum-likelihood estimation and various forms of regression [14]. These techniques identify parameter values by employing a single metric to align simulation dynamics with those of the real-world system. However, complex biological system dynamics are not well characterized by single metrics alone. They constitute many different types of interacting component, and encompass both positive and negative feedbacks. They are highly redundant: a single component can perform many functions and any one function can be performed by several components [15,16]. As such, calibration of a complex system simulation is fundamentally a multi-metric optimization problem; several metrics of a simulation's alignment with the biology must be simultaneously considered when evaluating putative parameter values. Consider, for example, cellular motility, which underlies many biological processes arising from cellular interaction. Which targets a given cell interacts with depends on both its speed and directional persistence; accurately modelling this process requires that metrics of both be considered.

In this paper, we position multi-objective optimization-based calibration (MOC, multi-objective calibration) as an important enabling technology for simulation-based biological investigation. Given its abstractive nature, a simulation undergoing calibration will not perfectly replicate all aspects of the biology. As such, putative simulation parameter value sets will exhibit trade-offs in their reproduction of aspects of the biology, excelling in some at the expense of others. In this context, a metric quantifying a simulation's capture of a specific aspect of the biology is termed an *objective*. Through the use of Pareto fronts (defined in §3), MOC explicitly tracks the collection of simulation parameter sets exhibiting optimal trade-offs between objectives. It is unknown if adopting baseline parameter values from different regions of the Pareto front will deliver fundamentally different conclusions from simulation-based experiments. The answer to this question is likely to be problem-specific, and the use of MOC allows this issue to be addressed by exposing a full range of Pareto-equivalent solutions.

Here we investigate multi-objective optimization, specifically the NSGA-II algorithm [17], in calibrating an established immunological simulation: ARTIMMUS (artificial murine multiple sclerosis simulation) [18]. ARTIMMUS simulates experimental autoimmune encephalomyelitis (EAE), a mouse model of multiple sclerosis [19,20]. It is a complex simulation, encompassing seven distinct cell populations that interact across five organs, and constituting 72 parameters. Its successful prior manual calibration renders it an effective test case for evaluating MOC's applicability to simulation calibration. We demonstrate the successful calibration of ARTIMMUS using five objectives (§4): a range of solutions to the calibration problem, offering optimal trade-offs against calibration objectives, are generated. Furthermore, we demonstrate that conclusions drawn from a simulation-based experiment can vary depending on exactly which calibration solution is adopted (§5). Hence, different calibration solution parameter values can vary downstream conclusions, highlighting MOC's value in making these multiple solutions explicit. We show that MOC is equally applicable in generating simulation initial condition values: cellular population sizes as simulation launch. We proceed to demonstrate that MOC can identify parameter and initial condition values that deliver previously unknown simulation dynamics, highlighting its potential beyond this well-understood test case (§6). Lastly, we consider strategies for formulating stopping criteria for MOC, thereby preventing overfitting and wasted computational expense when apparent improvements in simulation calibration are probably due to stochastic sampling rather than genuinely superior parameter values (§7). We begin by introducing ARTIMMUS (§2) and the MOC methodology (§3).

2. A test bed for calibrating biological simulations

ARTIMMUS is an ABS of an EAE protocol wherein mice induced into autoimmunity undergo a natural recovery from disease and are thereafter resistant to disease re-induction [18,21,22]. ARTIMMUS was created, in part, to further probe the cellular interactions mediating this recovery [23,24]. It has been used to explore the mechanisms through which splenectomy, the removal of the spleen, a primary immune organ, exacerbates disease severity and predict the outcome of T-cell interaction-blocking drugs [18]. It was conceived through a collaboration of immunologists and computer scientists, and developed through a principled approach focusing on documenting how biological concepts are translated into computer code: the CoSMoS process [25]. It is written in the Java programming language.

ARTIMMUS has previously undergone a by-hand, manual calibration [11], and was shown to reflect the dynamics of the real-world disease [18]. The process demanded close collaboration between the simulation developer and an immunologist who informed the work, helping bridge biological data and concepts to simulation constructs and output. This manual calibration took two weeks, and entailed an iterative process through which simulation code and parameter value changes that might explain perceived discrepancies between simulation and biological system dynamics were identified and explored in turn. Those best aligning simulation with biological dynamics were adopted before repeating the process. This calibration approach is akin to a non-population, manual, greedy local search wherein the best immediate improvement is always adopted.

Despite delivering a well-calibrated result for ARTIMMUS, this calibration search strategy presents several potential pitfalls. It is entirely plausible that the manual search does not find the global optimum parameter set that best aligns simulation dynamics with those of the biological system. As a greedy search strategy, its result is highly dependent on the search's starting position, and complex landscapes where one parameter's influence on simulation dynamics critically depends on the values held by others are particularly challenging. The existence of multiple solutions to the calibration problem can go entirely undetected. Lastly, manual calibration is time-consuming, and ABS's stochastic nature further compounds these challenges. It is these issues that collectively motivated the present automated MOC approach.

Here we provide a brief summary of EAE and ARTIMMUS to aid understanding of the sections that follow; a comprehensive description may be found in the supplementary materials of [18]. Figure 1*a* provides an abstract overview of the major cell types involved in EAE, and their relationships to one another. EAE is induced through injection of neuronal fragments which are internalized by dendritic cells (DCs) which then direct the growth of a T-cell population (CD4Th1, abbreviated to Th1) targeting these fragments. These Th1 cells enter the central nervous system (CNS), where they stimulate CNS-resident macrophages into secreting TNF-α, which in turn damages neurons. The resultant neuronal fragments are internalized by further populations of DCs, which direct further Th1 activities, perpetuating the autoimmune cycle. Recovery from autoimmunity is through the actions of two populations of regulatory T cell, CD4Treg and CD8Treg cells, so named as they regulate the activities of other T cells. The natural life cycle of a Th1 cell results in its eventual death and internalization by DCs, which derive fragments therefrom and direct the growth of CD4Treg and CD8Treg cells targeting the Th1 cell population. CD4Tregs play an essential role in facilitating the development of CD8Treg cells. CD8Treg cells can directly kill Th1 cells, interrupting their natural life cycle and preventing the perpetuation of autoimmunity. Th2 cells directly compete with Th1 cells, as both arise from a common progenitor and they each perform downstream activities that promote their own development. The reduced severity of the autoimmune environment arising from the action of CD8Treg cells favours the growth of Th2 cells over Th1 cells, which do not directly harm neurons and hence do not contribute to this autoimmune process. Figure 1*b* shows a time series graph of T-cell population sizes in ARTIMMUS.

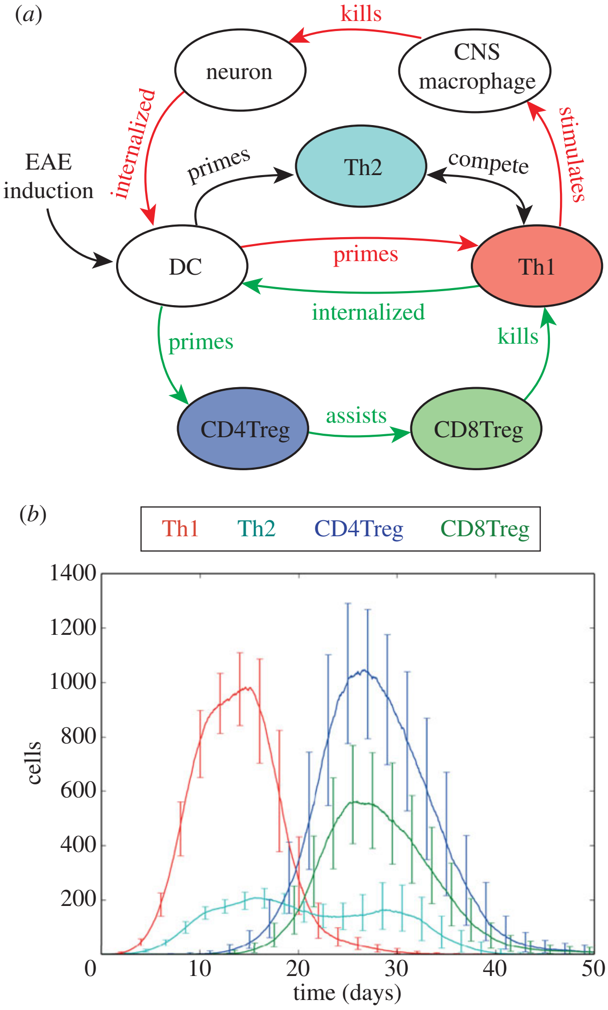


Figure 1. The ARTIMMUS used as a test case for evaluating MOC. (*a*) The major cell types represented in ARTIMMUS and their key influences on one another. Red and green arrows, respectively, indicate activities that perpetuate autoimmunity and mediate recovery. (Adapted from [11].) (*b*) The baseline dynamic of ARTIMMUS, depicting four T-cell population sizes over time. The simulation behaviour depicted here forms a calibration target for MOC in §4. Lines correspond to like-coloured cells in (*a*); these colours are maintained throughout the article. Error bars capture 90% of the data derived from 500 simulation executions; time series lines indicate median population sizes at each time point.

3. Multi-objective calibration methodology

We present here an overview of the MOC concept, detailing how we employ multi-objective optimization technology to calibrate simulation parameters and initial conditions. A graphical overview is supplied in figure 2.

Firstly, we define the desired (*target*) ARTIMMUS dynamics (figure 2*a*). In this paper, targets are expressed as peak cell population sizes, the times at which those peaks occur, or the cell population sizes at a given time. Target dynamics might represent known biological results to be reproduced, or hypothetical outcomes of interest. In this study, we adopt the dynamics of a previous manual calibration of ARTIMMUS, so as to evaluate MOC on a

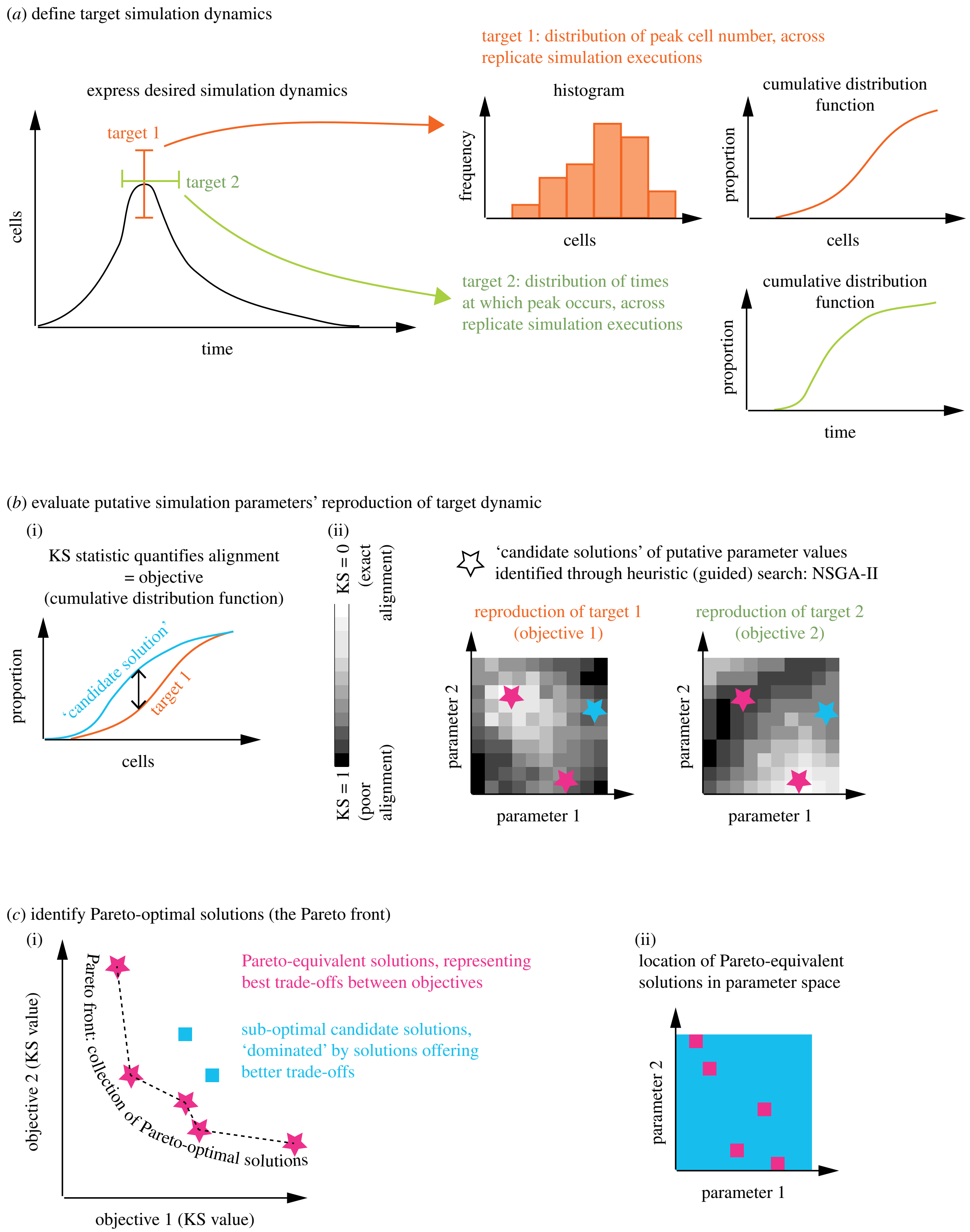

(a) define target simulation dynamics
express desired simulation dynamics
target 1
target 2
cells
time
target 1: distribution of peak cell number, across replicate simulation executions
histogram
frequency
cells
cumulative distribution function
proportion
cells
target 2: distribution of times at which peak occurs, across replicate simulation executions
cumulative distribution function
proportion
time
(b) evaluate putative simulation parameters' reproduction of target dynamic
(i)
KS statistic quantifies alignment = objective (cumulative distribution function)
'candidate solution'
target 1
proportion
cells
(ii)
KS = 0 (exact alignment)
KS = 1 (poor alignment)
'candidate solutions' of putative parameter values identified through heuristic (guided) search: NSGA-II
reproduction of target 1 (objective 1)
reproduction of target 2 (objective 2)
parameter 2
parameter 1
parameter 2
parameter 1
(c) identify Pareto-optimal solutions (the Pareto front)
(i)
Pareto front: collection of Pareto-optimal solutions
Pareto-equivalent solutions, representing best trade-offs between objectives
sub-optimal candidate solutions, 'dominated' by solutions offering better trade-offs
objective 2 (KS value)
objective 1 (KS value)
(ii)
location of Pareto-equivalent solutions in parameter space
parameter 2
parameter 1

Figure 2. Overview of the multi-objective calibration (MOC) concept. (*a*) The desired (target) simulation dynamics are defined as distributions (only two shown): the desired distributions of peak cell number and the times at which these occur. Distributions are depicted as histograms, or the corresponding cumulative distribution functions describing the proportion of samples in the distribution (*y*-axis) that hold a given value or less (*x*-axis). (*b*) The capacity for putative simulation parameter (only two shown) values, termed *candidate solutions*, to reproduce target dynamics is evaluated. The Kolmogorov–Smirnov (KS) statistic quantifies the difference between target and a given candidate solution's simulation performance (i); this metric is termed an *objective*. By sampling and evaluating regions of parameter space, we identify those that provide good alignment with a given objective, illustrated through greyscale heat maps (ii). No single region of parameter space maximizes performance against all objectives (only two shown); there exist inherent trade-offs. A heuristic (guided) search strategy, NSGA-II, is employed to strategically sample parameter space. (*c*) *Solutions* representing optimal trade-offs in performance against each objective are identified, collectively termed the *Pareto front* (i). These solutions are *Pareto equivalent* (pink): no solution has been found that represents an improvement in one objective without a worsening in another. Suboptimal candidate solutions are discarded (blue). Pareto-equivalent solutions may reside in disparate regions of parameter space (ii).

well-understood problem; thereafter we employ MOC to obtain hypothetical dynamics not known possible *a priori*. We note that many other aspects of simulation performance can constitute target dynamics, depending on the context and simulation being calibrated. The expression of targets as distributions reflects the stochastic nature of biological systems and ABSs, wherein repeat experiments can yield slightly different results.

MOC seeks to identify parameter values that best align simulation with target dynamics. As such, we define metrics, termed *objectives*, that quantify the alignment between the two. As illustrated in figure 2*b*(i), we employ the non-parametric Kolmogorov–Smirnov (KS) statistic in our objectives, which quantifies the difference in target and simulation dynamics for a given set of simulation parameter values. Rather than contrasting the medians of two distributions, as many statistics do, the KS statistic quantifies the biggest distance between two distributions' cumulative distribution functions. As such, its use here facilitates the calibration of a distribution's shape, not simply its median or mean. We consider this a strength of our approach; as may be seen in the sections that follow, MOC is capable of reproducing distributions of behaviour, not simply averages. Each set of simulation parameter values is termed a 'candidate solution', and its corresponding simulation performance is evaluated against each objective individually. By evaluating many candidate solutions, we identify regions of parameter space providing close alignment with target dynamics (figure 2*b*(ii)). Importantly, the regions that satisfy each objective differ. In practice, it is computationally intractable to fully explore parameter space as suggested by the heat maps in this figure, particularly when many parameters are investigated. Instead, a heuristic (guided) search strategy is employed that samples parameter space, evaluates performance and decides from where to extract the next candidate solutions based on the results. In this study, we employ NSGA-II as our guided search engine [17], but we believe other multi-objective optimization technologies could be successfully substituted. NSGA-II maintains a population of candidate solutions, and employs (heavily abstracted) principles of genetic recombination, mutation and natural selection to generate and evaluate successive generations of superior candidate solutions. Hence, NSGA-II is an iterative algorithm. We refer readers to [17] for more detail on NSGA-II. Here we have employed the 'inspyred' python module NSGA-II implementation.

We identify those candidate solutions that constitute optimal trade-offs in performance against each objective, referred to simply as *solutions* (figure 2*c*). The set of solutions is termed the *Pareto front*. These solutions are *Pareto equivalent*: no solution has been found that offers an improvement in one objective without a worsening in another. Pareto-equivalent solutions may reside in disparate regions of parameter space, and the ability to recognize this is a key strength of MOC. Though these regions of parameter space may be Pareto equivalent for the given target simulation behaviour, they could yield very different behaviours when subjected to further downstream experimentation, and as such lead to different simulation-borne conclusions. In this study, we investigate this phenomenon for a given experiment in ARTIMMUS.

We note that it is possible to derive a great many targets and objectives for complex system simulations. Increasing the number of objectives increases the difficulty of the calibration problem, and the computational resource required to address it; in the field of optimization this is known as the 'curse of dimensionality'. Hence, employing fewer, uncorrelated objectives is considered good practice: it encourages the identification of good quality solutions, while minimizing the resources required to do so.

3.1. Selecting candidates from the Pareto front

Upon completion MOC delivers a Pareto front of Pareto-equivalent solutions, representing optimal trade-offs between the calibration objectives. Deciding which solution adopts as the baseline simulation parameter values is an application-specific problem. For this study we have developed a function, $\Lambda(c)$, which assesses candidate solution c against the criteria below. We select the candidate with the lowest Λ value when presenting the results of calibration below. Λ is calculated as follows.

Let Ω represent the set of calibration objectives, and $\mathrm{KS}_o(c), o \in \Omega$ the corresponding KS score for candidate c on objective o. $\overline{\mathrm{KS}}(c)$ represents the mean objective score for candidate c. The Λ score is calculated as

$$\Lambda(c) = \alpha \cdot \overline{\mathrm{KS}}(c)^2 + \sum_{o\in\Omega}(\mathrm{KS}_o(c) - \overline{\mathrm{KS}}(c))^2. \tag{3.1}$$

Low Λ scores are achieved through low mean objective KS scores, and balanced KS scores across all objectives. α specifies the relative importance of these two components. When $\alpha = 1$, both measures contribute equally to Λ. Lower mean KS scores are prioritized with $\alpha > 1$ and vice versa. We employ $\alpha = 1$ throughout. We note that Λ is unit-less, and as such is not explicitly reported here; it is used only to extract one candidate solution from a Pareto front, presented as the chief result of calibration in the results that follow.

4. Successful re-calibration of ARTIMMUS

We demonstrate MOC by re-calibrating ARTIMMUS, taking as target dynamics those of the previous manually calibrated simulation dynamics [18]. As these dynamics are known to be obtainable, and at least one set of parameter values that produces them is known, we are able to evaluate MOC's performance.

With five objectives MOC successfully reproduced the manually calibrated ARTIMMUS dynamics, as demonstrated in figure 3. The objectives used were

— the peak Th1 cell population size (figure 3*b*),
— the time at which the peak occurred (figure 3*c*),
— the Th2 population size at 30 days (figure 3*d*), and
— the peak population sizes of both CD4Treg and CD8Treg cells (figure 3*e*,*f*).

The corresponding target distributions of values are also shown in figure 3.

Each candidate solution generated by NSGA-II was assessed through 200 replicate simulation executions. The target distributions against which candidates are contrasted are derived from 500 replicates generated with the previous manual-calibration parameter values. The

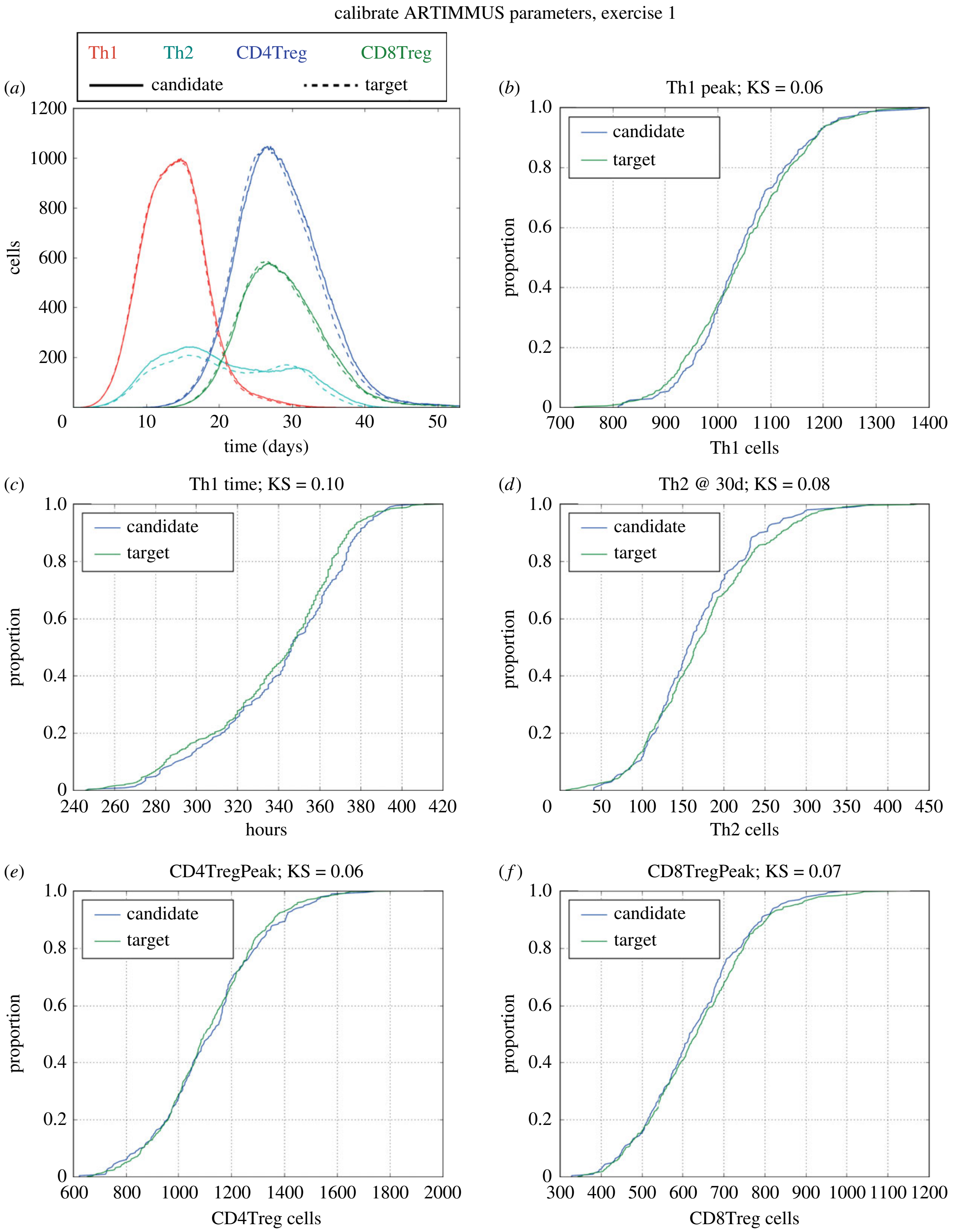


Figure 3. Multi-objective calibration (MOC) successfully re-calibrates ARTIMMUS parameters against five objectives. The best solution's, i.e. that with the lowest Λ score, target simulation dynamics are shown. The solution dataset comprises 200 simulation replicates; the target comprises 500. (*a*) T-cell population sizes over time, for both target (dotted line) and solution (solid line). The median values from each dataset at the given point in the time series are plotted. (*b*–*f*) Cumulative distribution functions showing alignment of solution and target distributions of values for each objective, with titles giving KS values. These graphs show the distribution of calibration target values obtained in each dataset: the *y*-axis indicates the proportion of items in the distribution holding a value less than or equal to the corresponding *x*-axis value. Objectives are (*b*) peak CD4Th1 population size cell; (*c*) time at which this peak occurs; (*d*) CD4Th2 population size at 30 days; (*e*) peak CD4Treg population size; and (*f*) peak CD8Treg population size. These data represent the first of three independent recalibration experiments.

manual-calibration's replicates need be executed once only and stored; they do not change. By contrast, assessment of candidates is computationally costly because so many are generated; a figure of 200 replicates per candidate was selected to strike a balance between experimental sensitivity and computational cost. A previous analysis of parametric perturbation in ARTIMMUS established that contrasting distributions comprising 200 replicate executions was sufficient to detect 'small' changes in two-thirds of simulation behaviour metrics, and 'medium' in the remainder [11]. Hence, we consider 200 replicates to offer sufficient sensitivity in differentiating candidate performances. These effect size categories arise from the analysis's use of the Vargha–Delaney *A test* [26], which provides interpretation guidelines. For reference, the *A test* is a non-parametric effect magnitude test representing the probability that a randomly selected member of one distribution is larger than a randomly selected member of the other. An *A test* score of 0.5 indicates that the two distributions are indistinguishable (using this test). Values of 1 and 0 indicate no overlap in the two distributions. A single calibration exercise required around 5 days on a dedicated computational cluster able to execute 120 simulations simultaneously; each single simulation replicate takes around 2–10 min to execute, depending on the parameter values used.

We have successfully applied MOC to both ARTIMMUS parameter values and initial conditions, but focus here on the former. Initial condition calibration results are reported in the electronic supplementary material. Calibration was performed over eight ARTIMMUS parameters which all pertain to presentation of substances to T cells, particularly Th1 and Th2 cells, and their resultant development. The biology captured in these parameters is outlined in the electronic supplementary material, figure S1, and we note that a thorough understanding of this biology is not required to appreciate our results. These parameters were selected for the reasons that ascertaining their values experimentally would be challenging and they all relate to a critical aspect of the biology: the perpetuation of autoimmunity and (for some) its amelioration (as Treg cell development is also directed by DCs). Hence, by successfully calibrating parameter values that are highly influential on simulation dynamics we demonstrate MOC's potential. Parameters were given a constrained range of values that the MOC process could assign, being zero to twice their manually calibrated range, as shown in table 1. In exploring the space of putative parameter values, NSGA-II maintained a population of 64 candidate solutions which were subject to genetic recombination and mutation [17] over 32 generations of natural selection, wherein only the best 64 solutions (i.e. those on or near the Pareto front) were retained in the successive generation.

This calibration exercise was repeated three times for both parameters and initial conditions. Figure 3 shows the solution with the lowest Λ score from one such parameter calibration. The remaining two are shown in the electronic supplementary material, figures S2 and S3. The calibrated simulation dynamics closely resemble the target distributions in all cases. The three parameter calibration exercises generated, respectively, Pareto fronts constituting 82, 87 and 112 Pareto-equivalent solutions. The ranges of parameter values represented across the Pareto fronts' solutions in each independent calibration exercise are shown in figure 4, as are the baseline manually calibrated values. In all but one case, the baseline parameter value sat within the range of non-outlier MOC-derived values, the exception being *Th1_diff80* in exercise 3. Hence, we conclude that MOC is an effective means of calibration: it has repeatedly reproduced ARTIMMUS dynamics that were known possible, and has identified similar solutions, in the form of parameter values, that do so.

Table 1. The ARTIMMUS parameters (*a*) and initial conditions (*b*) subject to calibration, their baseline (manually calibrated) values, and the lower and upper bounds of values they may be assigned during MOC.

(*a*) parameters calibrated

parameter	baseline value	lower bound	upper bound
APC_immatureDuration	48	0	96
APC_matureDuration	110	0	220
APC_phagocytosisToPeptide	0.02	0	0.04
CNSM_MBPExpressionProbability	0.2	0	0.4
DCT1_cytokineSecretionRate	10	0	20
DC_T2CytokineRatio	0.17	0	0.34
Th1_diff00	0.05	0	0.1
Th1_diff80	0.85	0	1.0

(*b*) initial conditions calibrated

initial condition	baseline value	lower bound	upper bound
numTh	40	0	80
numCD4Treg	30	0	60
numCD8Treg	30	0	60
numCNS	500	0	1000
numCNSMacrophage	75	0	150
numDC	10	0	20
numDCCNS	40	0	80
numDCSpleen	100	0	200

Next, we investigated how the space of ARTIMMUS parameter values relates to the space of successful target dynamic reproductions, i.e. trade-offs in objective values. We find statistically significant ($p < 0.01$) differences between calibration exercises' distributions of calibrated parameter values for seven of eight parameters (figure 4). This corresponds to 19 of 24 (79%) pairwise comparisons. Furthermore, 75% (18/24) pairwise comparisons register a KS $\geq$ 0.3. For context, a KS value of 1.0 indicates no overlap between two distributions. By contrast, this degree of variation is not observed in Pareto fronts' objective values, depicted in figure 5. Here, we instead find statistically significant differences in only 27% (4/15) of pairwise calibration comparisons, and only 27% (4/15) of comparisons register KS $\geq$ 0.3. We find no evidence of objectives that are harder to calibrate than others; the smallest objective values are less than 0.05 in all cases, and the median objective values all lie under 0.17.

Together, these data suggest a redundancy in the ability for parameter values to deliver particular objective scores. This corresponds to a landscape wherein parameter values mapped to objective values is relatively flat, as a wide range of ARTIMMUS parameter values deliver relatively similar objective scores. The results of using MOC to calibrate ARTIMMUS initial conditions are reported in the electronic supplementary material, section S1, and figures S4, S5 and S6. They are qualitatively identical to our findings in calibrating parameters and support the conclusions drawn here.

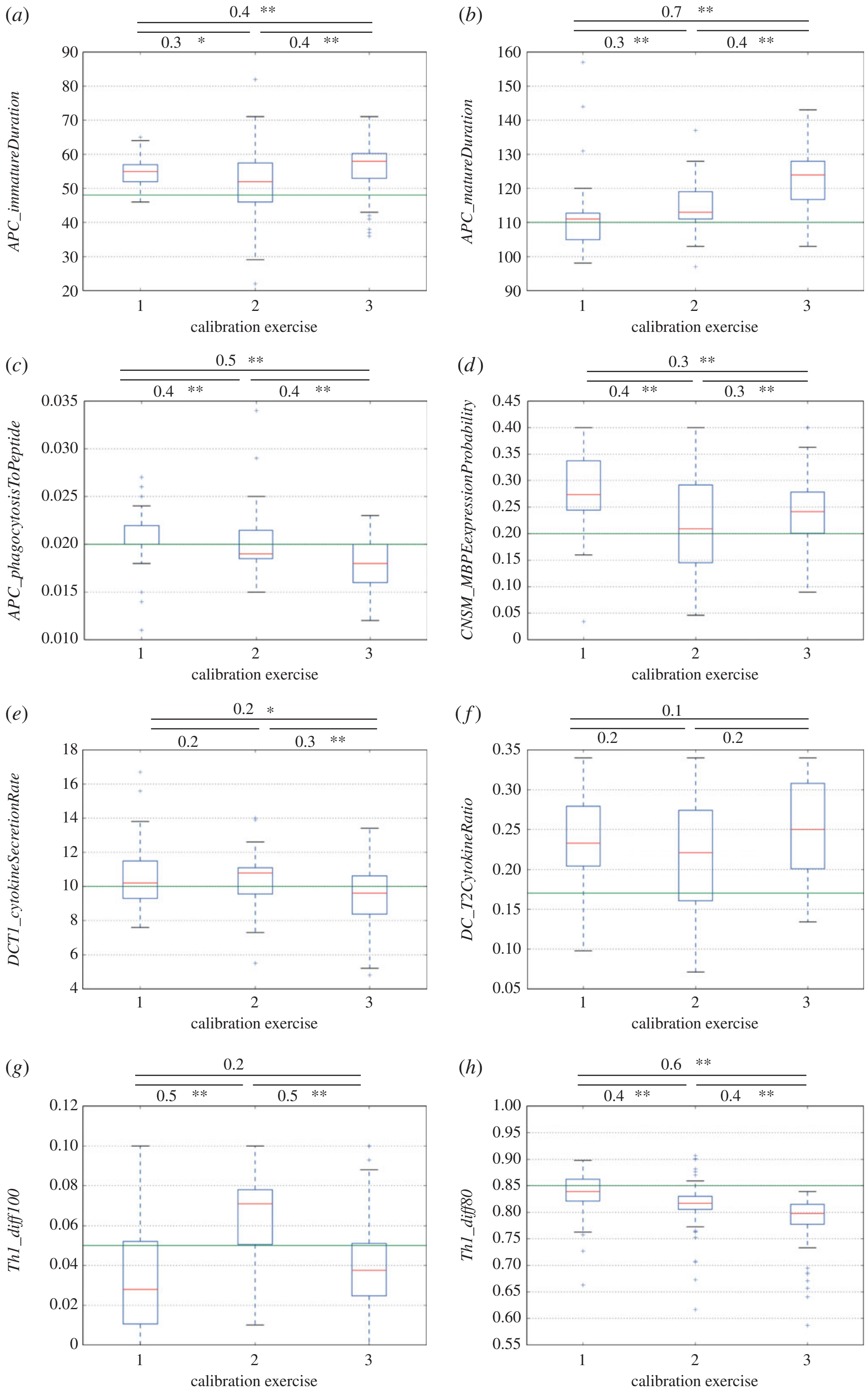


Figure 4. Automated re-calibration of ARTIMMUS parameters delivers solutions approximating the original manually calibrated parameter values. Box plots are shown for each of three independent calibration exercises. The horizontal solid green line represents the manually calibrated parameter values. Calibration was performed over five objectives: the peak population sizes of Th1, CD4Treg, CD8Treg cells, the time at which the Th1 population peaks and the number of Th2 cells at 30 days. Parameters subject to calibration are listed in table 1; see the electronic supplementary material, figure S1, for an explanation of their operation in ARTIMMUS. Values shown above each plot are the KS scores between distributions, shown to one significant figure; the associated p-values are: $^{*}p < 0.01$ and $^{**}p < 0.001$. Outliers in boxplots are defined as lying beyond the first or third quartiles by 1.5 times the interquartile range. (Online version in colour.)

An obvious question is: why does MOC not deliver any perfectly calibrated solutions, wherein all objective scores are 0.0? The best solutions, determined by their minimal Λ values, in each calibration exercise are shown in table 2. Objective KS values ranged from 0.05 to 0.14 (and from 0.03 to 0.12 for initial conditions). We attribute the inability to deliver a perfect calibration to the stochastic nature of ARTIMMUS, wherein 200 replicate executions for a given candidate yields sufficient variation so as to deliver objective KS scores of greater than or equal to 0.05. There is

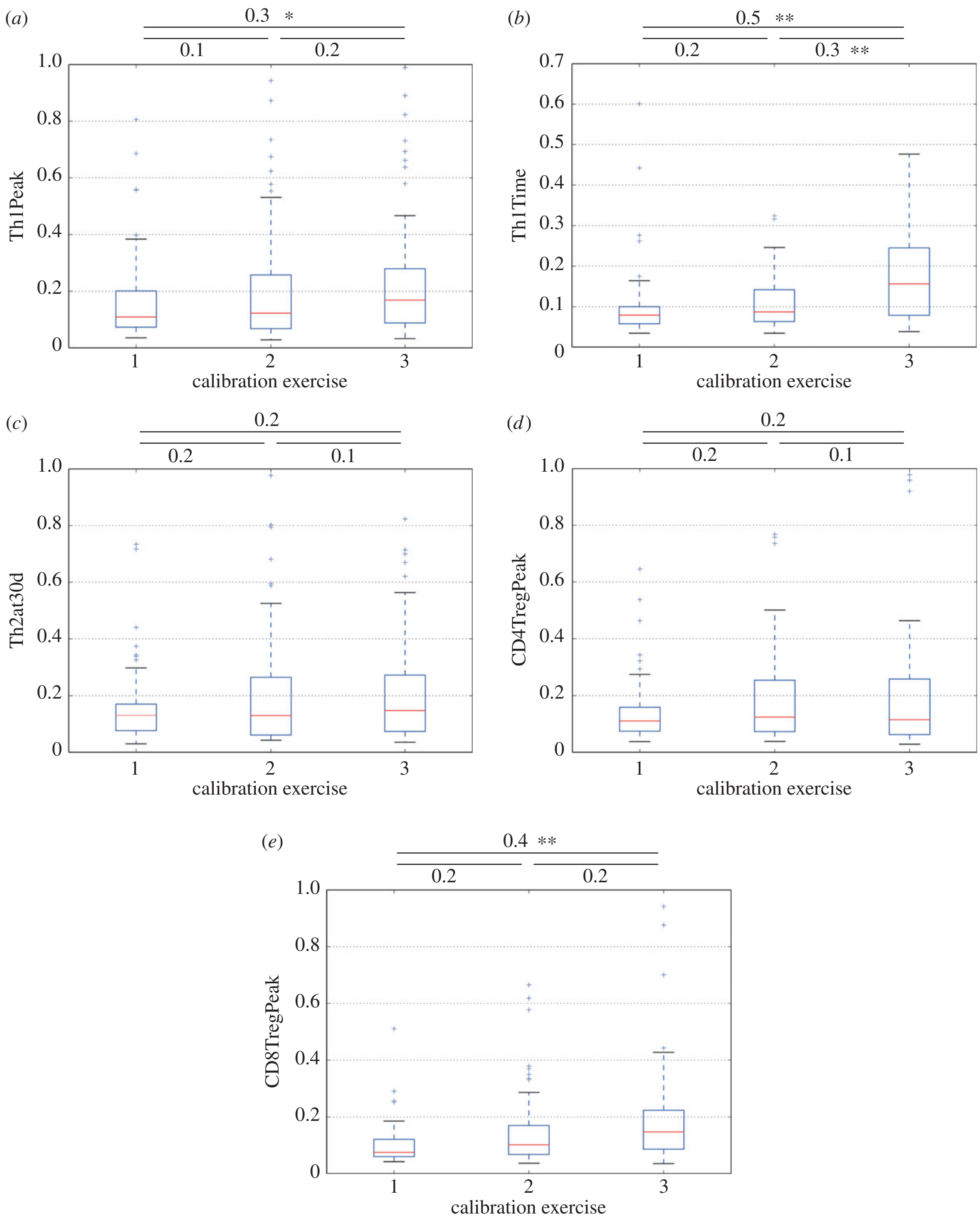


Figure 5. The range of objective values that constitute the Pareto front derived through MOC re-calibration of ARTIMMUS. Box plots are shown for each of three independent calibration exercises. These objective values correspond to the Pareto front and associated ARTIMMUS parameter values of figure 4. Calibration was performed against five objectives: (*a*) the peak population size of Th1 cells; (*b*) the time at which this occurred; (*c*) the number of Th2 cells at 30 days; (*d*) the peak population size of CD4Treg cells; (*e*) the peak population size of CD8Treg cells. Statistical and boxplot formatting are as in figure 4. (Online version in colour.)

a risk that improvements in objective KS values that are already so small cannot be confidently attributed to an actual improved simulation calibration, as opposed to stochastic variation between simulation replicates. Section 7 explores a method for terminating the MOC process on the premise that further effort will not deliver better quality solutions.

These data collectively highlight the challenges in exactly calibrating (i.e. KS = 0.0) simulations to several objectives simultaneously. As such, we consider in the next section the implications on experimental results of adopting baseline simulation values from different extremes of the Pareto front.

5. Scientific significance of imperfect calibration

As demonstrated above, MOC delivers a host of solutions to a given calibration problem, each representing an optimal trade-off in calibration criteria (figure 2). It falls on the simulation developer to decide which baseline parameter values to

Table 2. The best solution, being that with the lowest Λ value, arising from each of three independent calibration exercises. Shown are each of the five objective KS values. We independently investigated the calibration of both ARTIMMUS parameters (*a*) and initial conditions (*b*). High-quality calibrations, as indicated by low KS values, were obtained in all cases.

	objective KS value				
calibration exercise	Th1Peak	Th1Time	Th2at30d	CD4TregPeak	CD8TregPeak
(*a*) calibration on parameters					
1	0.06	0.10	0.08	0.06	0.07
2	0.08	0.06	0.06	0.05	0.07
3	0.05	0.08	0.14	0.08	0.05
(*b*) calibration on initial conditions					
1	0.06	0.08	0.04	0.03	0.06
2	0.04	0.08	0.10	0.11	0.12
3	0.06	0.06	0.06	0.07	0.05

adopt in subsequent experimentation. There is a risk that, while calibration solutions lying in different regions of parameter space give rise to Pareto-equivalent solutions, they do not behave in a consistent manner when further experiments are performed. In such a case, a simulation-based experiment would lead to different conclusions depending on which calibration result was adopted as the baseline. In this section, we investigate the extent to which this phenomenon holds.

The manually calibrated ARTIMMUS was previously used to elucidate the effect of removing a central immune organ, the spleen (a *splenectomy*), in EAE-induced animals [18]. Previous experiments had demonstrated that splenectomy in rats prior to the induction of EAE increased the mortality rate and hampered recovery [27]. Simulating splenectomy in ARTIMMUS revealed the spleen as a primary site for the generation of autoimmunity-combating CD4Treg and CD8Treg cells. The reduced Treg populations resulting from the spleen's removal prior to EAE induction were unable to completely abrogate the autoimmunity-inducing Th1 populations, allowing for their re-expansion, and thus facilitating increased disease severity and relapses.

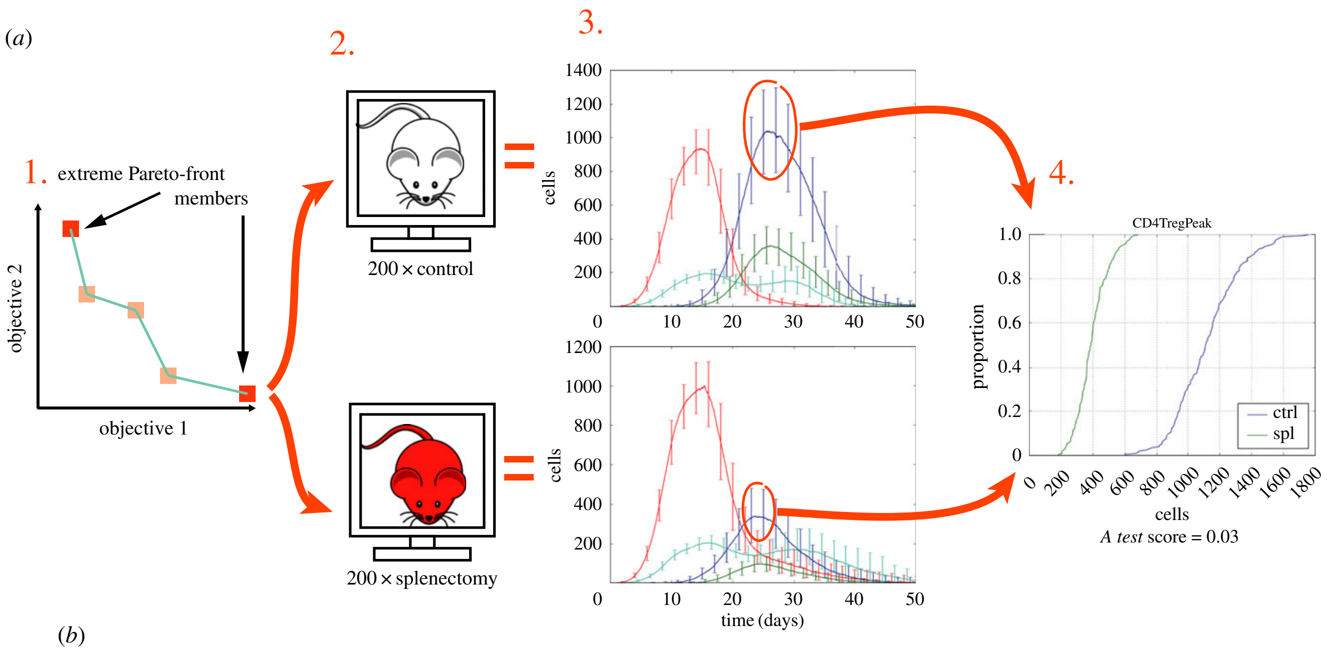
Here we explore whether the results of splenectomy in ARTIMMUS differ when baseline parameter values are adopted from disparate extremes of the Pareto front. The experimental procedure is highlighted in figure 6. First, Pareto front solutions representing the extreme values, both low and high, of objective KS measures are identified. These solutions represent extremes in the range of simulation dynamics encapsulated within the Pareto front. For each solution 200 simulation replicates are performed for both control and splenectomy groups. Key performance indicators (KPIs) are extracted from the resultant distributions of 200 simulation executions in each group. The performance indicators used are identical to those of the original ARTIMMUS splenectomy experiment [18]: the peak population sizes for each T-cell population in the simulation, the times at which these peaks are reached, and the number of Th1 cells remaining at day 40 (giving a total of nine). For each KPI, the distributions of values obtained for control and splenectomy groups are contrasted using the Vargha–Delaney *A test* [26], as per the original experiment [18]. This procedure is repeated for each of the three calibration exercises reported in §4. The resultant *A test* scores are shown in figure 6*b*. Also shown, for context, are the *A test* scores of the original ARTIMMUS experiment [18].

Broadly speaking, the splenectomy results generated by Pareto-equivalent solutions are consistent with one another, and with the original experiment. There are exceptions, however, wherein differences in *A test* scores reported for solution and the original experiment differed substantially: g23c60 in exercise 1, and g6c35 and g30c58 in exercise 2. These differences occurred for 'Th1 @ 40d', 'Th2 peak' and 'Th2 Time' KPIs. Of interest, three of these solutions were obtained from the region of the Pareto front where alignment with target Th2 peak population size was poorest. In the case of g23c60 and g6c35, exercises 1 and 2, respectively, the parameter values where sufficient to return Th1 population size at 40 days to control group levels, despite the splenectomy ($A = 0.58$ and 0.56; 0.5 indicates no difference). This is significant, as the principal conclusion of the original experiment was that splenectomy reduces Treg population sizes to levels unable to suppress Th1 cell populations and abrogate autoimmunity. The time-series T-cell population dynamics of both these solutions under control and splenectomy are shown in the electronic supplementary material, figure S9. In both cases, the peak Th1 population sizes are smaller than in the original experiment (figure 6), and the Th2 population sizes are substantially larger. Based on this, we hypothesize that, despite reduced Treg population sizes resulting from splenectomy, the altered balance between Th1 and Th2 populations which compete with one another is sufficient to abrogate the Th1 population at day 30 in these solutions.

Supporting the notion that solutions' results are relatively consistent, the direction of change in solutions' KPIs resulting from splenectomy differs from the original experiment in only a minority of cases. Furthermore, this occurs only in KPIs for which the original experiment reports a comparatively small change between splenectomy and control, the largest being in exercise 2 when the original experiment reports a change of $A = 0.66$, which was not interpreted as significant.

We have conducted the same investigation on Pareto-equivalent solutions generated under the three independent initial condition calibration exercises (electronic supplementary material, section S1). Detailed analysis is reported in the electronic supplementary material, section S2 and figure S10; briefly, divergences between the initial condition

(*a*)



(*b*)

calibration exercise 1, Vargha-Delaney A test scores

	Th1Peak		Th1Time		Th2at30d		CD4TregPeak		CD8TregPeak				
response	g31c32 (KS=0.04)	g11c33 (0.81)	g29c37 (0.03)	g10c13 (0.60)	g31c6 (0.03)	g23c60 (0.73)	g30c0 (0.04)	g23c22 (0.65)	g10c13 (0.04)	g23c22 (0.51)	orig	diff	d.c.
Th1 Peak	0.70	0.73	0.65	0.67	0.67	0.66	0.71	0.67	–	–	0.62	0.11	
Th1 Time	0.53	0.51	0.55	0.42	0.53	0.52	0.52	0.52	–	–	0.47	0.08	y
Th2 Peak	0.60	0.74	0.66	0.66	0.67	0.67	0.61	0.62	–	–	0.66	0.08	
Th2 Time	0.47	0.53	0.68	0.46	0.51	0.56	0.53	0.47	–	–	0.58	0.11	y
CD4Treg Peak	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	–	–	0.00	0	
CD4Treg Time	0.23	0.24	0.26	0.26	0.20	0.28	0.20	0.24	–	–	0.21	0.07	
CD8Treg Peak	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	–	–	0.00	0	
CD8Treg Time	0.24	0.24	0.29	0.27	0.23	0.25	0.25	0.26	–	–	0.23	0.06	
Th1 at 40d	0.98	0.96	0.92	0.89	0.96	0.58	0.96	0.94	–	–	0.95	0.37	

calibration exercise 2, Vargha-Delaney A test scores

response	g30c25 (KS=0.03)	g6c35 (0.94)	g30c34 (0.03)	g9c63 (0.33)	g15c14 (0.04)	g30c58 (0.98)	g17c61 (0.04)	g14c54 (0.77)	g9c56 (0.04)	g14c54 (0.67)	orig	diff	d.c.
Th1 Peak	0.63	0.69	0.66	0.66	0.71	0.65	0.65	0.66	0.68	–	0.62	0.09	
Th1 Time	0.53	0.49	0.55	0.47	0.50	0.53	0.54	0.51	0.48	–	0.47	0.08	y
Th2 Peak	0.66	0.67	0.65	0.70	0.68	0.39	0.65	0.61	0.70	–	0.66	0.27	y
Th2 Time	0.56	0.51	0.63	0.63	0.50	0.31	0.54	0.48	0.54	–	0.58	0.27	y
CD4Treg Peak	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	–	0.00	0.01	
CD4Treg Time	0.27	0.30	0.23	0.19	0.20	0.21	0.17	0.22	0.29	–	0.21	0.09	
CD8Treg Peak	0.00	0.00	0.00	0.01	0.00	0.00	0.00	0.00	0.00	–	0.01	0.01	
CD8Treg Time	0.27	0.26	0.23	0.17	0.21	0.20	0.22	0.23	0.28	–	0.23	0.06	
Th1 at 40d	0.96	0.56	0.94	0.91	0.94	1.00	0.97	0.97	0.82	–	0.95	0.39	

calibration exercise 3, Vargha-Delaney A test scores

response	g5c39 (KS=0.03)	g23c10 (0.99)	g28c47 (0.04)	g13c21 (0.48)	g11c6 (0.04)	g5c39 (0.82)	g24c62 (0.03)	g23c10 (0.98)	g23c40 (0.04)	g23c10 (0.94)	orig	diff	d.c.
Th1 Peak	0.65	0.65	0.63	0.70	0.68	–	0.65	–	0.70	–	0.62	0.08	
Th1 Time	0.43	0.49	0.49	0.42	0.47	–	0.47	–	0.52	–	0.47	0.05	y
Th2 Peak	0.67	0.64	0.70	0.68	0.67	–	0.67	–	0.67	–	0.66	0.04	
Th2 Time	0.59	0.51	0.52	0.50	0.47	–	0.62	–	0.57	–	0.58	0.11	y
CD4Treg Peak	0.07	0.02	0.00	0.00	0.00	–	0.00	–	0.00	–	0.00	0.07	
CD4Treg Time	0.30	0.35	0.21	0.24	0.26	–	0.20	–	0.19	–	0.21	0.14	
CD8Treg Peak	0.04	0.01	0.00	0.00	0.00	–	0.00	–	0.00	–	0.00	0.04	
CD8Treg Time	0.27	0.40	0.17	0.26	0.27	–	0.20	–	0.21	–	0.23	0.17	
Th1 at 40d	0.93	0.83	0.97	0.90	0.92	–	0.97	–	0.95	–	0.95	0.12	

Figure 6. Do different regions of MOC's Pareto front of solutions give rise to different results in subsequent experimentation? (*a*) An overview of the experimental procedure. (1) Pareto-front members representing objective value extremes are identified (only two objectives are shown in the example). (2) The simulation parameters represented by such members are adopted in performing a control and splenectomy experiment, with 200 replicate simulations in each group. (3) Key performance indicators are extracted from the resultant distributions of simulation dynamics; indicated here is the peak CD4Treg population size within each individual simulation. (4) Performance indicators are statistically contrasted for splenectomy and control experiments. These statistics are examined across different Pareto-front members, thereby gauging the extent to which experimental results critically depend on which Pareto-equivalent parameter values are adopted in simulation. (*b*) Tables: columns represent extreme Pareto-front solutions, defined as having either the highest or lowest KS value for each of the five objectives used in calibration (see §4). The objective KS value scores are shown in parentheses. Only the first occurrence of each solution is shown, with subsequent entries indicated by '—'. Rows indicate the difference between control and splenectomy simulations based on each solution according to key indicators of simulation behaviour, as measured by the Vargha–Delaney *A test* [26]. The original *A test* scores for the manually calibrated simulation are shown (orig), as is the biggest difference in *A test* score observed between manually and automatically calibrated simulations (diff). Values highlighted in red represent four differences in candidate and original *A test* scores that are notably larger than differences observed elsewhere. 'd.c.' indicates 'direction change', where there exists at least one candidate for which the *A test* score lay on the other side of 0.5 from the original, indicating that the distribution of values under splenectomy increased in the original experiment but decreased for the candidate (or vice versa).

solution and the original experiments were smaller than reported here for parameters. We take this to indicate that the initial parameters investigated were less influential on simulation behaviour than the parameters investigated here.

In summary, the conclusions that would be drawn from adopting baseline parameter values from disparate Pareto-equivalent solutions are mostly, but not completely, consistent with one another and with the original splenectomy experiment. There were two notable exceptions, and they underscore the importance of considering the range of simulation performances that satisfy a calibration exercise. Making these explicit through Pareto fronts is a strength of the MOC approach. It remains important to, where possible, further evaluate Pareto-equivalent solutions in the context of domain knowledge and expertise, which might have ruled out the two exceptions noted above, as the Th2 population size is abnormally large compared with the Th1 population. Where this is not possible, where no grounds to discard some Pareto-equivalent solutions exist, we advise that experiments are performed in replicate, adopting a wide range of calibration solutions, and that conclusions are drawn after taking stock of the full range of results generated. This is particularly important if quantitative, rather than qualitative, results are sought; our present data show more divergence between calibration solutions and original experiment in the quantitative case.

6. Multi-objective calibration delivers previously unseen disease phenotypes

In §4, MOC successfully reproduced simulation dynamics known to exist by virtue of a prior manual calibration. To further demonstrate MOC's generality and utility, we now derive simulation dynamics not known to exist *a priori*.

The baseline behaviour of ARTIMMUS constitutes a period of autoimmunity followed by recovery, reflecting typical biological disease [21,22]. However, disease susceptibility and severity vary considerably between mouse strains and between mice within a given strain [28,29]. Furthermore, depletion or incapacitation of CD4Treg and CD8Treg cells leads to exacerbated disease symptoms [30,31]. Here we investigate the capacity for ARTIMMUS to reproduce persisting disease symptoms of varying severity. To reflect potential genetic differences between mouse strains, we calibrate over initial conditions specifying cell population sizes, and a parameter controlling the efficiency of Th1 killing by CD8Treg cells; together comprising nine variables. In this experiment, we are implicitly investigating whether variation in these basal population sizes and the efficiency of the CD8Treg–Th1 killing pathways could explain the differences in autoimmune phenotypes observed between mouse strains and individuals therein.

Three persisting disease severities are investigated, ranging from mild to severe. These are captured by defining the distribution of Th1 cells remaining at 60 days as a target for calibration, captured as a Gaussian distribution. Mild, moderate and severe disease are represented with mean (μ) and standard deviation (σ) values of $\mu = 50$ and $\sigma = 10$, $\mu = 200$ and $\sigma = 100$, and $\mu = 500$ and $\sigma = 200$, respectively. To ensure an aggressive onset of autoimmunity, consistent with animal models, a second calibration target distribution of $\mu = 1000$ and $\sigma = 200$ Th1 cells at 15 days is employed.

Each persisting autoimmunity severity is independently calibrated three times, representatives of which are shown in figure 7 (the remainder are shown in the electronic supplementary material, figures S11, S12 and S13). Automated calibration successfully delivers the required median number of cells in most cases, with KS ≤ 0.2 in six of the nine calibrations. However, the spread of the 'Th1 cells at 60 days' distribution for mild persisting disease is notably less well calibrated, with all three calibrations delivering KS > 0.3.

Together, these data support the general applicability of MOC to problems where a simulation's ability to deliver a desired dynamic is not known *a priori*. These data also suggest that the heterogeneity in disease severities observed in experimental animals could be attributed to differences in basal population sizes and regulatory pathway efficiency.

7. When to stop multi-objective calibration

A key consideration in any optimization task is the stopping criteria. For MOC, underpinned by the NSGA-II optimization algorithm, this equates to determining when to stop calibration.

Overfitting describes the case where the simulation being calibrated starts to capture the noise in the target distributions, rather than the trends those distributions represent. This is a particular issue when target distributions do not contain many samples, as might be the case if they represent biological experiments (figure 8*a*). For example, studies involving experimental animals can require their sacrifice to collect data. As such, it is considered unethical (and is practically cumbersome) to collect hundreds of samples, and 5–10 is more typical. These smaller sample sizes are unlikely to perfectly capture the underlying distribution that would emerge if thousands of samples were available. Overfitting is said to have occurred when the calibrated simulation better reflects these 5–10 samples than their underlying distribution, as illustrated in figure 8*b*.

A common strategy in single-objective (not MOC, which is multi-objective) problems for determining when to terminate an optimization process is to segregate the available data into two parts, termed 'training' and 'validation' datasets. The training dataset is used as normal to search for improved solutions, akin to MOC's target data. The validation dataset is used as an independent check for overfitting of solutions to the training dataset. Such a case of overfitting is depicted in figure 8*c*. Both the training and validation data roughly reflect the underlying distribution from which they were sampled. The candidate solution more closely resembles the training dataset than either the underlying distribution or the validation dataset; hence, it is overfitted. As illustrated in figure 8*d*, in the earlier stages of optimization successive candidate solutions that better capture the training dataset will also better capture the validation data. It is only when overfitting starts to occur that performance against the validation data worsens while performance against training data continues to improve. It is at this point that the optimization process is best terminated.

MOC is, however, a multi-objective optimization problem, and it is unclear in the literature how this overfitting detection strategy ought to be applied. We propose here a novel strategy for detecting overfitting in multi-objective problems based on co-membership of solutions to both training and validation dataset Pareto fronts (P_{t} and P_{v}), maintained

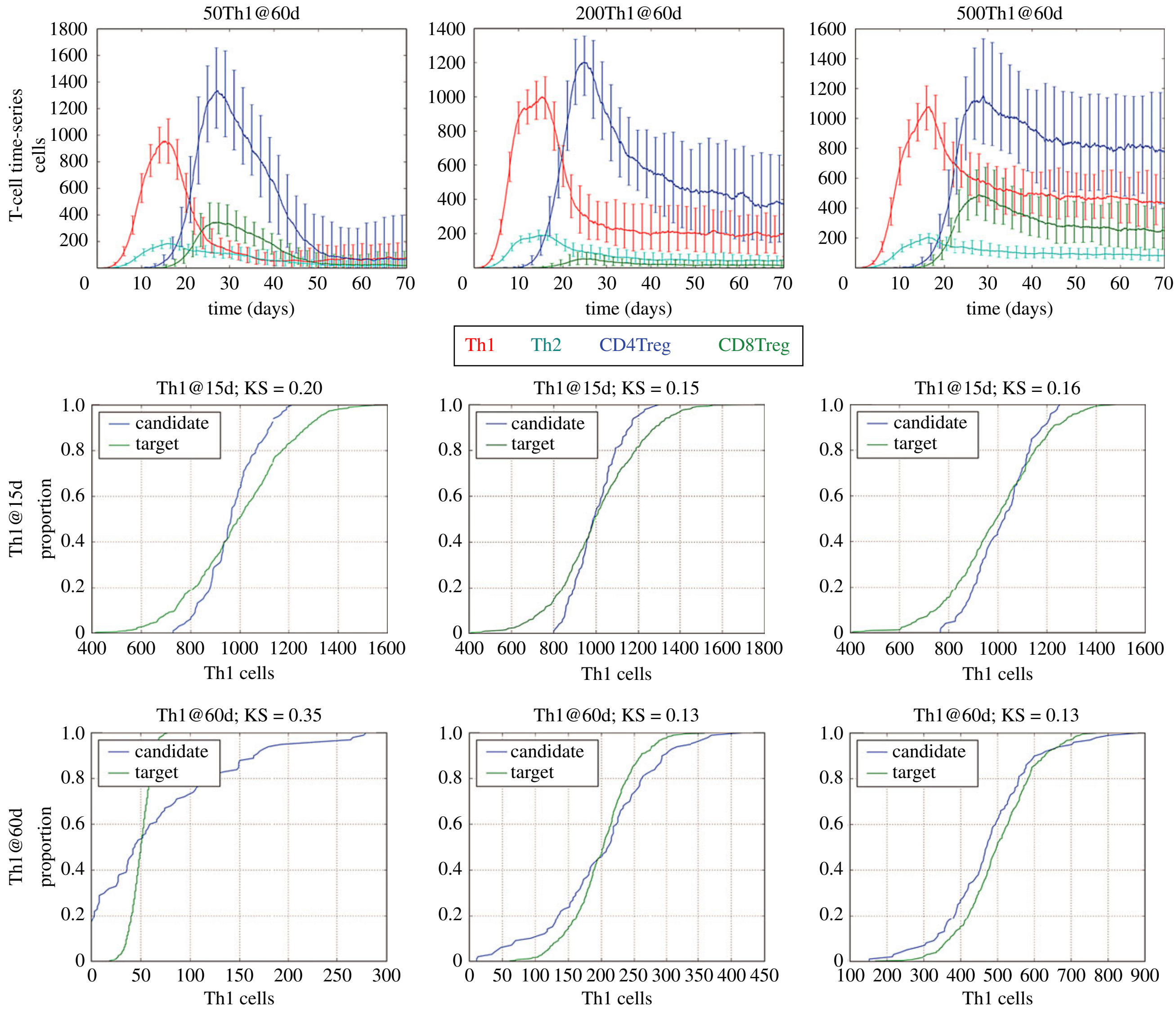


Figure 7. Employing MOC to discover parameter and initial condition values delivering simulation dynamics not known to exist *a priori*: persisting autoimmune states of varying severity. Three severities are explored, represented as columns. They are a mean of 50, 200 or 500 Th1 cells at 60 days (with standard deviations of 10, 100 and 200, respectively). A second objective is employed in all cases, 1000 Th1 cells at 15 days, which drives the establishment of autoimmunity. Each severity is calibrated in three independent experiments, and shown here are the solutions exhibiting the lowest Λ values from a representative calibration of each experiment. The first row of graphs depicts the median T-cell time series. The second row shows the candidate's performance against an objective of 1000 Th1 at 60 days (s.d. = 200). The last row depicts the second objective, the (respective) number of T cells at 60 days.

throughout the calibration process (figure 8*e*). The overfittedness at a given point in the optimization process is reflected in the proportion of P_t members that are not members of P_v. The following algorithm performs the calculation:

```
$m \leftarrow 0$
for all $i \in P_t$ do
    if $i \in P_v$ then
        $m \leftarrow m + 1$
    end if
end for
return $1 - (m/\mathrm{size}(P_t))$
```

A proportion of 0 indicates that all training dataset Pareto solutions are also members of the validation Pareto front. At the other extreme, a value of 1 indicates that the training dataset Pareto front has been completely overfitted, as none of its members are Pareto optimal with respect to the validation dataset. A threshold level of overfitting at which the optimization process (i.e. MOC) is to be terminated can be selected by the simulation experimenter.

We investigated different overfitting thresholds for MOC termination in the three ARTIMMUS parameter recalibration exercises reported in §4. An additional 214 simulation replicates using manually calibrated parameter values were acquired to use as a validation dataset, constituting a 70–30 (500–215) training–validation data split. The validation dataset Pareto front for each iteration of the MOC algorithm (generation) was determined, and the overfittedness calculated. Figure 9*a* shows how, as MOC progresses, the proportion of overfitted candidate solutions on the training (target) dataset increases for each of the three calibration exercises. Figure 9*b* shows the point at which MOC would have been terminated should a given overfittedness threshold have been selected. Had we employed an overfittedness termination threshold of 0.5, wherein half of the training dataset Pareto front is overfitted, calibration would have

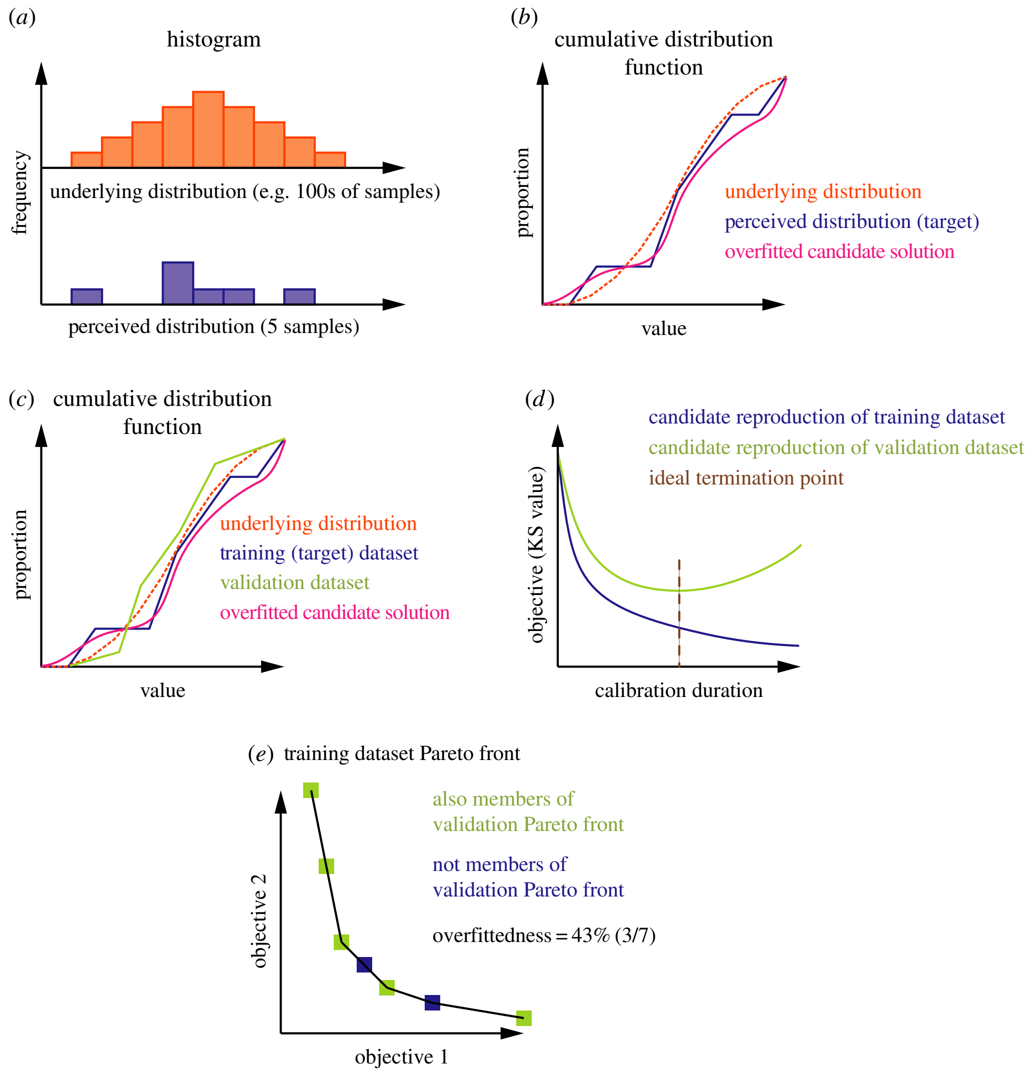


Figure 8. Terminating MOC when overfitting occurs. Overfitting describes the case when solutions generated by an optimization process, e.g. MOC, better resemble the target data than the underlying distribution from which they were drawn. (*a*) In many contexts, such as animal experiments, only limited samples of a phenomenon can be obtained. The samples will broadly, but not exactly, reflect the underlying distribution. (*b*) An overfitted candidate solution more closely resembles the target data than the underlying distribution from which the target data were drawn. Detecting this is difficult because the true underlying distribution cannot be absolutely known. (*c*) A common strategy in single-objective optimization problems is to divide the available data into two: a training dataset and a validation dataset. The training dataset is used as the target in obtaining successively better quality solutions. The validation dataset is used as an independent check. Overfitting is detected when solutions more closely resemble the training dataset than the validation dataset. This is illustrated in (*d*), where early solutions generally offer improved performance against both datasets. It is only in later stages that solutions so closely reflect the target dataset that they diverge from the validation dataset. This is when the process should be stopped. (*e*) Overfitting can be detected in multi-objective optimization, such as MOC, by maintaining Pareto fronts of optimal solutions against both training and validation data independently. The degree of overfitting is reflected in the proportion of training data Pareto-front solutions that are not members of the validation data Pareto front.

terminated at generation 14, 15 or 23 (for exercises 1, 2 or 3, respectively) instead of 32. Given that each of these calibration exercises required around 7 days to complete on a dedicated computing cluster, this speed-up is substantial. We note that these combined training and validation datasets constitute 714 data points, considerably exceeding what might be obtained from real biological experiments. We anticipate that with fewer data points overfitting will occur sooner in the MOC process.

8. Discussion

Simulation represents a powerful tool to advance the investigation of biological systems, particularly when used in tandem with traditional approaches. As more complex biological systems become the subject of simulation a challenge in their calibration emerges: complex biological systems cannot be characterized by single metrics alone. There exist technologies capable of identifying parameter values that align simulation dynamics with some desired target, but these operate on single metrics. Even in cases where parameter values can be ascertained experimentally, seemingly avoiding the need for calibration, the abstract nature of simulation can complicate their direct adoption. Here, we have demonstrated how biological ABS parameter values can be derived using multi-objective optimization, an approach we have termed MOC. Multi-objective optimization algorithms find solutions to problems simultaneously described by more than one metric. In MOC the desired characteristics of the simulation, which can represent either established biological data to be reproduced or some desired

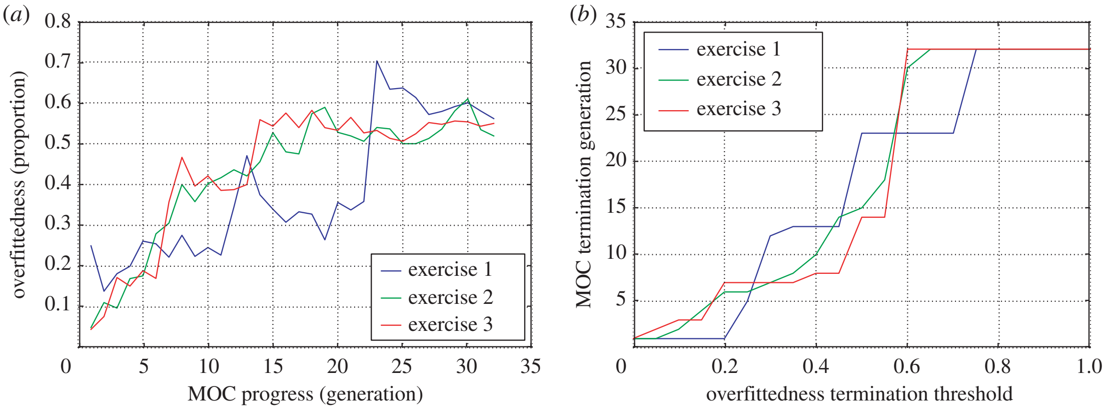


Figure 9. Empirical results for detecting overfitting in MOC, and when to terminate the process accordingly. We generated a validation dataset using ARTIMMUS's previous manually calibrated parameter values, and retrospectively analysed how overfitted MOC solutions would have been on the three MOC exercises reported in §4. (*a*) The overfittedness, defined as the proportion of MOC Pareto-front solutions that are not also members of a similar Pareto front maintained for the validation data, at each MOC generation. (*b*) The generation at which MOC would have been terminated for a given overfittedness threshold value.

hypothetical simulation outcome, are expressed as distributions. Importantly, several such characteristics can be expressed, and MOC identifies those sets of parameter values that deliver optimal trade-offs against each.

We evaluated MOC on a well-understood simulation, using it to reproduce a previous manual calibration effort and therein delivering a solution that was known to be possible. The ARTIMMUS was used, which simulates a mouse multiple sclerosis disease model [18]. MOC delivered around 90 unique parameter value combinations, each of which provided an optimal trade-off in performance against the five target ARTIMMUS characteristics specified. This range of possible calibration solutions was unknown *a priori*; the previous manual calibration of ARTIMMUS having delivered only one such solution [11]. It would ordinarily fall on the simulation user to select one solution (set of parameter values) to adopt as a baseline for subsequent simulation experimentation. We investigated the significance of selecting solutions representing different extremes of trade-offs in delivering target simulation characteristics. A previous experiment with ARTIMMUS determined that removing the spleen, an important immune system organ, resulted in exacerbated autoimmune symptoms. The results of re-performing this experiment with different MOC solutions adopted as baseline parameter values were broadly, but not absolutely, similar. Hence, adopting different calibration solutions can lead to different experimental conclusions. It a strength of MOC that this range of solutions is made explicit. Where possible, we recommend that MOC solutions be evaluated against biological data to discard those that represent biologically unrealistic parameter values or behaviours. Where this is not possible, we advocate performing experiments in replicate using multiple MOC solutions such that the full range of possible results be established before conclusions are drawn.

We demonstrated MOC in deriving simulation behaviours that were not known possible *a priori*: varying degrees of persisting autoimmunity in ARTIMMUS. MOC can be applied to both parameters and initial conditions, at the same time, as demonstrated in these calibration exercises. We do not consider simulation parameter values and initial conditions as independent; a poor selection of initial condition values coupled with appropriate parameter values can still fail to deliver the desired simulation dynamic. MOC's successful delivery of these previously unknown simulation dynamics presents an interesting use case for MOC. It could be used to identify which parameters, and hence components and pathways, need to be manipulated to resolve a simulated disease state, therein highlighting candidate therapeutic targets. Furthermore, for disease simulations that incorporate potential interventions, MOC can be used to determine optimal intervention strategies that exploit synergies between several treatment options.

We surmise that MOC can support model selection and development. Accurately simulating a biological system requires both an appropriate model of the biology and appropriate parameter values for that model. There typically exist several options for how to represent a biological concept in simulation, the most suitable of which is often unclear. Models must strike a balance between including sufficient complexity to accurately reflect the biology's dynamics while remaining sufficiently simplistic to offer insight. The unsuccessful calibration of a given model of the biology can lead to two conclusions: first, that the calibration process was simply unsuccessful in finding a solution that does exist, a risk we argue is greatly lessened through MOC; or second, that the model is incapable of replicating the biological dynamics in question. In this latter case, MOC can inform simulation design, where a succession of putative models can be evaluated until calibration is successful. The possibility of directly applying MOC to the space of biological abstractions, rather than parameter values, is intriguing, though extremely challenging technically. Here, MOC would search for which cells were represented, and how. This would encompass their interactions with one another, opting to ignore some found to be irrelevant to the biological phenomenon of interest or vice versa. The level of detail through which molecular secretions and expressions where represented could also be determined: is a variable expression level necessary, or does simply 'present' versus 'not' suffice? The challenge herein lies in building an ABS infrastructure capable of capturing all these possibilities and allowing the automated optimization process to manipulate them. The aforementioned point still applies—for each possible model, the space of parameter values must also be investigated, as an accurate reflection of biology requires both an appropriate model and corresponding parameter values. Hence, MOC would be applied in a nested fashion, first over the space of biological representations, and therein over the space of parameter values for each model.

Although our present investigation has employed an ABS, MOC is applicable to other simulation paradigms also, such as ordinary differential equations (ODEs). Application to non-stochastic simulations, such as ODEs, requires significantly less computational power, as there is no need to obtain simulation replicates in assessing a candidate solution's fitness. We note that, from our experience in building them, not all biological simulations are as computationally costly to execute and calibrate as ARTIMMUS. Each MOC exercise has taken up to a week of time on a dedicated computational facility. In this regard, terminating the MOC process when a threshold level of overfitting is detected is pertinent (see figure 8). Overfitting was detected in all three of our ARTIMMUS parameter recalibration exercises, and selecting a threshold of 0.5, wherein half of the MOC solutions at a given point no longer represent optimal performance trade-offs in an independent test, could as much as halve the computational effort required.

The ability to detect overfitting in a multi-objective context is a novel contribution of this work. Although a common strategy for stopping a single-objective optimization process, it was previously unclear how to deploy this strategy in a multi-objective context [32]. There is another condition under which we feel it pertinent to terminate the MOC process. The goal of MOC is to find parameter values yielding simulation dynamics that closely resemble some target. As this alignment increases, and differences in solutions' simulation performances reduce, it is possible that seemingly better alignments in fact represent sampling artefacts arising from the stochastic simulation, rather than genuinely superior parameter values. We note that detecting this in a statistically robust manner is challenging, and as such we highlight it as potential further work.

This work fits within the context of a wider framework for supporting complex system simulation, the CoSMoS framework [25]. CoSMoS advocates explicitly recording, typically through graphical modelling [33], how real-world concepts are translated into computer code, and the implicit assumptions therein. In this context, MOC can help in relating simulation results to biological data. The case where a distribution of results emerges from a given biological experiment, even to the point where replicates or individuals within an experiment exhibit completely different outcomes, can be handled in MOC by defining bi-modal (or multi-modal) target distributions. A scenario wherein MOC unexpectedly delivers several distinct and unconnected simulation phenotypes, rather than a continuum of points on the Pareto front, is interesting. This either can suggest the existence of additional phenotypes to look for in the biology or, if this can be ruled out, suggests instead that the model being calibrated fails to accurately capture the biology. This latter case is an example of how MOC could drive simulation design and development, as covered above. Related work on supporting the link of simulation to biology proposes the construction of an argument wherein a claim such as 'this simulation is an adequate representation of the biology' is supported by explicitly cited evidence [34]. In this context, application of MOC can raise confidence that appropriate parameter and initial condition values have been identified. The range of possible values can be contrasted against biological literature and data, excluding those deemed implausible. Subsequent simulation experiments can be performed in replicate with those that remain, therein highlighting the full range of results that are plausible in the absence of a better reason to rule out particular parameter values. We argue that drawing conclusions from this nature of simulation experimentation, and making explicit the full range of parameter values that satisfy the calibration problem, leads to more robust conclusions.

In summary, our novel application of multi-objective optimization in MOC presents the multi-objective optimization community with a new field of application, and one we feel has considerable scope for growth. Importantly, it provides fundamental support for a critical aspect of simulation-based biological experimentation: identifying parameter values and initial conditions that align simulations with a complex target behaviour.

Authors' contributions. M.N.R. conceived and designed the study, carried out all data generation and analysis reported herein, and drafted the manuscript. K.A. and J.T. conceived the study and drafted the manuscript. L.M.R. conceived the study.

Competing interests. We have no competing interests.

Funding. M.N.R. is supported by the David & Judith Coffey Life Lab. J.T. is partly funded by the Royal Society.

Acknowledgements. We thank Dr Fabian Held, Prof. Susan Stepney and Dr Simon Poulding for engaging discussions on statistics.

References

1. Bauer AL, Beauchemin CA, Perelson AS. 2009 Agent-based modeling of host-pathogen systems: the successes and challenges. *Inf. Sci.* **179**, 1379–1389. (doi:10.1016/j.ins.2008.11.012)
2. Harris TH *et al.* 2012 Generalized Lévy walks and the role of chemokines in migration of effector CD8 + T cells. *Nature* **486**, 545–548. (doi:10.1038/nature11098)
3. An G, Mi Q, Dutta-Moscato J, Vodovotz Y. 2009 Agent-based models in translational systems biology. *Wiley Interdiscip. Rev. Syst. Biol. Med.* **1**, 159–171. (doi:10.1002/wsbm.45)
4. Pienaar E, Dartois V, Linderman JJ, Kirschner DE. 2015 In silico evaluation and exploration of antibiotic tuberculosis treatment regimens. *BMC Syst. Biol.* **9**, 79. (doi:10.1186/s12918-015-0221-8)
5. Thorne BC, Bailey AM, DeSimone DW, Peirce SM. 2007 Agent-based modeling of multicell morphogenic processes during development. *Birth Defects Res. C Embryo Today* **81**, 344–353. (doi:10.1002/bdrc.20106)
6. Senior AM, Charleston MA, Lihoreau M, Buhl J, Raubenheimer D, Simpson SJ. 2015 Evolving nutritional strategies in the presence of competition: a geometric agent-based model. *PLoS Comput. Biol.* **11**, e1004111. (doi:10.1371/journal.pcbi.1004111)
7. Kitano H. 2002 Systems biology: a brief overview. *Science* **295**, 1662–1664. (doi:10.1126/science.1069492)
8. Kitano H. 2002 Computational systems biology. *Nature* **420**, 206–210. (doi:10.1038/nature01254)
9. Cosgrove J, Butler J, Alden K, Read M, Kumar V, Cucurull-Sanchz L, Timmis J, Coles M. 2015 Agent-based modelling in systems pharmacology. *CPT: Pharmacometrics Syst. Pharmacol.* **4**, 615–629. (doi:10.1002/psp4.12018)
10. Dong C, Martinez GJ. 2010 T cells: the usual subsets. Nature Reviews Immunology Poster. See www.nature.com/nri/posters/tcellsubsets.
11. Read M, Andrews PS, Timmis J, Kumar V. 2012 Techniques for grounding agent-based simulations in the real domain: a case study in experimental autoimmune encephalomyelitis. *Math. Comput.*

Model. Dyn. Syst. **18**, 67 – 86. (doi:10.1080/13873954.2011.601419)

12. Calvez B, Hutzler G. 2006 Automatic tuning of agent-based models using genetic algorithms. In *Multi-agent-based simulation VI* (eds JS Sichman, L Antunes), pp. 41 – 57. Heidelberg, Germany: Springer.
13. Fabretti A. 2012 On the problem of calibrating an agent based model for financial markets. *J. Econ. Interact. Coord*. **8**, 277 – 293. (doi:10.1007/s11403-012-0096-3)
14. Burnham KP, Anderson DR. 2002 *Model selection and multimodel inference: a practical information-theoretic approach*, 2nd edn. New York, NY: Springer-Verlag.
15. Cohen IR. 2004 *Tending Adam's garden: evolving the cognitive immune self*. London, UK: Elsevier Academic Press.
16. Kitano H. 2004 Biological robustness. *Nat. Rev. Genet*. **5**, 826 – 837. (doi:10.1038/nrg1471)
17. Deb K, Pratap A, Agarwal S, Meyarivan T. 2002 A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput*. **6**, 182 – 197. (doi:10.1109/4235.996017)
18. Read M, Andrews PS, Timmis J, Williams RA, Greaves RB, Sheng H, Coles M, Kumar V. 2013 Determining disease intervention strategies using spatially resolved simulations. *PLoS ONE* **8**, e80506. (doi:10.1371/journal.pone.0080506)
19. Baxter AG. 2007 The origin and application of experimental autoimmune encephalomyelitis. *Nat. Rev. Immunol*. **7**, 904 – 912. (doi:10.1038/nri2190)
20. Pachner AR. 2011 Experimental models of multiple sclerosis. *Curr. Opin. Neurol*. **24**, 291 – 299. (doi:10.1097/WCO.0b013e328346c226)
21. Kumar V, Sercarz E. 2001 An integrative model of regulation centered on recognition of TCR peptide/MHC complexes. *Immunol. Rev*. **182**, 113 – 121. (doi:10.1034/j.1600-065X.2001.1820109.x)
22. Kumar V. 2004 Homeostatic control of immunity by TCR peptide-specific Tregs. *J. Clin. Invest*. **114**, 1222 – 1226. (doi:10.1172/JCI23166)
23. Greaves RB, Read M, Timmis J, Andrews PS, Butler JA, Gerckens BO, Kumar V. 2013 In silico investigation of novel biological pathways: the role of CD200 in regulation of T cell priming in experimental autoimmune encephalomyelitis. *Biosystems* **112**, 107 – 121. (doi:10.1016/j.biosystems.2013.03.007)
24. Williams RA, Greaves R, Read M, Timmis J, Andrews PS, Kumar V. 2013 In silico investigation into dendritic cell regulation of CD8Treg mediated killing of Th1 cells in murine experimental autoimmune encephalomyelitis. *BMC Bioinform*. **14**(Suppl. 6), S9. (doi:10.1186/1471-2105-14-S6-S9)
25. Bown J, Andrews PS, Deeni Y, Goltsov A, Idowu M, Polack FAC, Sampson AT, Shovman M, Stepney S. 2012 Engineering simulations for cancer systems biology. *Curr. Drug Targets* **13**, 1560 – 1574. (doi:10.2174/138945012803530071)
26. Vargha A, Delaney HD. 2000 A critique and improvement of the CL common language effect size statistics of McGraw and Wong. *J. Educ. Behav. Stat*. **25**, 101 – 132. (doi:10.2307/1165329)
27. Ben-Nun A, Ron Y, Cohen IR. 1980 Spontaneous remission of autoimmune encephalomyelitis is inhibited by splenectomy, thymectomy or ageing. *Nature* **288**, 389 – 390. (doi:10.1038/288389a0)
28. Levine S, Sowinski R. 1973 Experimental allergic encephalomyelitis in inbred and outbred mice. *J. Immunol*. **110**, 139 – 143.
29. Gold R. 2006 Understanding pathogenesis and therapy of multiple sclerosis via animal models: 70 years of merits and culprits in experimental autoimmune encephalomyelitis research. *Brain* **129**, 1953 – 1971. (doi:10.1093/brain/awl075)
30. Beeston T, Smith TR, Maricic I, Tang X, Kumar V. 2010 Involvement of IFN-g and Perforin, but not Fas/FasL interactions in regulatory T cell-mediated suppression of experimental autoimmune encephalomyelitis. *J. Neuroimmunol*. **15**, 91 – 97. (doi:10.1016/j.jneuroim.2010.07.007)
31. Kumar V, Stellrecht K, Sercarz E. 1996 Inactivation of T cell receptor peptide-specific CD4 regulatory T cells induces chronic experimental autoimmune encephalomyelitis (EAE). *J. Exp. Med*. **184**, 1609 – 1617. (doi:10.1084/jem.184.5.1609)
32. Dos Santos EM, Sabourin R, Maupin P. 2009 Overfitting cautious selection of classifier ensembles with genetic algorithms. *Inf. Fusion* **10**, 150 – 162. (doi:10.1016/j.inffus.2008.11.003)
33. Read M, Andrews PS, Timmis J, Kumar V. 2014 Modelling biological behaviours with the unified modelling language: an immunological case study and critique. *J. R. Soc. Interface* **11**, 20140704. (doi:10.1098/rsif.2014.0704)
34. Alden K, Andrews PS, Polack FAC, Veiga-Fernandes H, Coles MC, Timmis J. 2015 Using argument notation to engineer biological simulations with increased confidence. *J. R. Soc. Interface* **12**, 20141059. (doi:10.1098/rsif.2014.1059)