Genetic Algorithms for model refinement and rule discovery in a high-dimensional agent-based model of inflammation

Acknowledgements: This work was supported by National Institutes of Health grant 1RO1GM115839-01. Additionally, this research used high performance computing resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231, as well as resources provided by the Vermont Advanced Computing Core (VACC).

Author Summary: In this work, we utilize genetic algorithms (GA) to operate on the internal rule set of a computational of the human immune response to injury, the Innate Immune Response Agent-Based Model (IIRABM), such that it is iteratively refined to generate cytokine time series that closely match what is seen in a clinical cohort of burn patients. At the termination of the GA, there exists an ensemble of candidate model parameterizations which are validated by the experimental data;

Abstract

Introduction: Agent-based modeling frequently used modeling method for multi-scale mechanistic modeling. However, the same properties that make agent-based models (ABMs) well suited to representing biological systems also present significant challenges with respect to their construction and calibration, particularly with respect to the large number of free parameters often present in these models. The challenge of dealing with parameters is further exacerbated due to the fact that a great deal of phenotypic and clinical heterogeneity can be attributed to intrinsic genetic/epigenetic variation manifesting as functional parameter variation. As a result, effectively representing biological populations with ABMs requires dealing with very large multi-dimensional parameter spaces that exponentially increase the computational demands for their use. We have proposed that various machine learning (ML) and evolutionary computing approaches (such as genetic algorithms (GAs)) can be used to more effectively and efficiently deal with parameter space characterization; the current work applies GAs to the challenge of calibrating a complex ABM to a specific data set in a fashion that preserves the parameter spaces required to deal with biological heterogeneity.

Methods: This project uses a GA to fit a previously validated ABM of acute systemic inflammation, the Innate Immune Response ABM (IIRABM) to clinical time series data of systemic cytokine levels. The genome for the GA is a vector generated from the IIRABM's Model Rule Matrix (MRM), which is a matrix representation of not only the constants/parameters associated with the IIRABM's cytokine interaction rules, but also the existence of rules themselves. Capturing heterogeneity is accomplished by a fitness function that incorporates the sample value range ("error bars") of the clinical data.

Results: The GA-enabled parameter space exploration resulted in a set of putative MRM parameterizations which closely (though not perfectly) match the cytokine time course data used to design the fitness function. The number of non-zero elements in the MRM increases significantly as the model parameterizations evolve towards a fitness function minimum, transitioning from a sparse to a dense matrix. This results in a model structure that more closely resembles (at a superficial level) the structure of data generated by a standard differential gene expression experimental study, in that there are a small number of powerful causative correlations and a much larger number of weaker/less significant (individually) connections.

Conclusion: We present an HPC-enabled evolutionary computing approach that utilizes a GA to calibrate a complex ABM to clinical data while preserving biological heterogeneity. The integration of machine learning/evolutionary computing, HPC and multi-scale mechanistic modeling provides

a pathway forward to more effectively represent the heterogeneity of clinical populations and their data.

Introduction

Agent-based modeling is an object-oriented, discrete-event, rule-based, spatially-explicit, stochastic modeling method. Agent-based modeling is a powerful technique for representing biological systems; rules are derived from experimentally observed biological behaviors, and the spatially-explicit nature of the models give it an inherent ability to capture space/geometry/structure, which facilitates the ability of biomedical researchers to express and represent their hypotheses in an agent-based model (ABM) (1). ABM's have been used to study and model a wide variety of biological systems (2), from general purpose anatomic/cell-for-cell representations of organ systems capable of reproducing multiple independent phenomena (3, 4) to platforms for drug development (5, 6), and are frequently used to model non-linear dynamical systems such as the human immune system (7-10).

ABM's often have a large number of potentially free parameters, making a comprehensive calibration difficult (11-15) and significantly diminishing the utility of traditional parameter sensitivity analysis techniques (16, 17). These difficulties are compounded when considering the range of biological heterogeneity seen experimentally and clinically (9). There are two primary factors responsible for biological heterogeneity in experimental data sets: stochasticity and genetic variation among individuals.

It is well known in biology that the systemic response to identical perturbations in genetically identical individuals (i.e., mice) is governed according to some probability distribution. This small stochastic variability in response can propagate over time such that it ultimately leads to divergent phenotypes. As such, ABM's must incorporate some degree of randomness to simulate these behaviors. However, solely incorporating stochasticity into model rules is insufficient to capture the full range of bio-plausible model output – genetic variation among *in silico* test subjects must also be represented. The *in-silico* analogue to the human genome is the specific parameterization of an ABM's rule set. In order to represent a biological population, there must exist a range on each parameter within the rule-set parameterization.

In order to demonstrate this, we utilize a previously developed an ABM of systemic inflammation, the Innate Immune Response agent-based model (IIRABM). The IIRABM is a two-dimensional abstract representation of the human endothelial-blood interface. This abstraction is designed to model the endothelial-blood interface for a traumatic (in the medical sense) injury and does so by representing this interface as the unwrapped internal vascular surface of a 2D projection of the terminus for a branch of the arterial vascular network. The closed circulatory surface can be represented as a torus, and this two-dimensional area makes up the space that is simulated by the model. The spatial scale is not directly mapped using this scheme. This abstraction serves two primary purposes: to allow circumferential access to the traumatic injury by the innate immune system, and to incorporate multiple levels of interaction between leukocytes and tissue. The IIRABM utilizes this abstraction to simulate the human inflammatory signaling network response to injury; the model has been calibrated such that it reproduces the general clinical trajectories of sepsis. The IIRABM operates by simulating multiple cell types and their interactions, including endothelial cells, macrophages, neutrophils, TH0, TH1, and TH2 cells as well as their associated precursor cells. The simulated system dies when total damage (defined as aggregate endothelial cell damage) exceeds 80%; this threshold represents the ability of current medical technologies to keep patients alive (i.e., through organ support machines) in conditions that previously would have been lethal. The IIRABM is initiated using 5 parameters representing the size and nature of the injury/infection as well as a metric of the host's resilience– initial injury size, microbial invasiveness, microbial toxigenesis, environmental toxicity, and host resilience.

The IIRABM characterizes the human innate immune response through measurement of various biomarkers, including the pro-inflammatory and anti-inflammatory cytokines included in the IIRABM (18). At each time step, the IIRABM measures the total amount of cytokine present for all mediators in the model across the entire simulation. The ordered set of these cytokine measurement creates a high-dimensional trajectory through cytokine space that lasts throughout the duration of the simulation (until the *in silico* patient heals completely or dies. Prior analysis of these trajectories has shown that the aggregate output of the IIRABM behaves as a Random Dynamical System (RDS) with chaotic features (9) (in the sense that future simulation state can be sensitive to initial conditions). Simply put, an RDS is a system in which the equations of motion (in this case, the equations which give the aggregate cytokine value for the system at a specific instance in time) contain elements of randomness. A detailed discussion of this, and more formal definition, can be found in (19).

While the IIRABM successfully simulates the human immune response to injury at a high level (outcome proportions, time to outcome, etc.), it cannot always replicate specific cytokine time series with a sufficient degree of accuracy. In this work, we use Genetic Algorithms (GA) to operate on the IIRABM's rule set such that it can accurately simulate the cytokine time course and final outcomes for a serious burn injury. Cytokine time series were extracted via inspection from (20). In (20), Bergquist, et al, provide a variety of blood cytokine levels over 15 time points and 22 days for patients which exhibited severe burns over 50% of the surface area of their bodies. The authors observed a mortality rate of 50% for this category of injury.

Methods

A GA (21-23) is a population-based optimization algorithm that is inspired by biological evolution. In a GA, a candidate solution is represented by a synthetic 'genome,' which, for an individual, is typically a one-dimensional vector containing numerical values. Each individual in a genetic algorithm can undergo computational analogues to the biological processes of reproduction, mutation, and natural selection. In order to reproduce, two individual vectors are combined in a crossover operation, which combines the genetic information from two parents into their progeny.

In our computational models, we define an object, the Model Rule Matrix (MRM) which contains comprehensive information regarding the rules that govern the behavior of the computational model. In this scheme, specific rules are represented by rows in the matrix; each computationally relevant entity in the model is then represented by the matrix columns. As a simple example, the system of model rule equations for a single cell:

$$IL10_{t+1} = IL10_t + TNF_t$$
$$TNF_{t+1} = -IL10_t + IFNg_t$$

Would be represented by the matrix:

$$\begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

Where the first column is the IL10 column, the second column is the TNF column, and the third column is the IFN-γ column. We note that this is a simplified rule for illustration. The matrix is readily decomposable into a one-dimensional vector, upon which we can operate using genetic algorithms. The genome vector is then padded with an additional three parameters which govern the nature of the injury and how quickly damage spreads though tissue. This addition describes the component of the time evolution of the spatial distribution of a tissue injury that is independent of cytokine levels.

The number of rows in the matrix then is equal to the number of rules that it represents, and the number of columns is equal to the number of entities that could potentially contribute to the decision made by their associated rule. Using this scheme, cytokines produced by a given cell type are held fixed, while the stimuli that lead to the production of that specific cytokine are allowed to vary. This maintains a distinction between the cell and tissue types represented in the model throughout the MRM evolution from the GA.

The candidate genomes which comprise the rule set are then tested against a fitness function which is simply the sum of cytokine range differences between the experimental data and the computational model:

$$F = \sum_{i,t} \left|\max\left(c_{i,t}^{e}\right) - \max\left(c_{i,t}^{m}\right)\right| + k\left|R_e - R_m\right|,$$

where $c_{i,t}^{exp}$represents the normalized blood serum level of cytokine *i* at time point *t* from the experimental data, $c_{i,t}^{m}$represents the normalized blood serum level of cytokine *i* at time point *t* from the IIRABM, R_e represents the experimentally observed mortality rate, R_m represents the model-generated mortality rate, and *k* is an adjustable parameter to govern the importance of the mortality rate contribution to the fitness function. For the purposes of this work, we consider an optimal solution to be one that minimizes the above fitness function.

Candidate genomes are then selected against each other in a tournament fashion, with a tournament size of 2 [28, 29]. The tournament winners make up the breeding pool, and progenitor genomes are randomely selected and paired. We implement a variant of elitism in that, at the completion of the tournament, the least fit 10% of the candidate progenitors are replaced with the fittest 10% of candidate genomes from the precious generation. Progeny genomes are defined with a uniform crossover operation using a standard continuous formulation (24):

$$C_{1,i} = \beta P_{1,i} + (1-\beta)P_{2,i}$$
$$C_{2,i} = \beta P_{2,i} + (1-\beta)P_{1,i}$$

Where $C_{1,i}$ is the value for gene *i* in child 1, P is the value for gene *i* in parent 1, and β is a random floating-point number between 0 and 1. After breeding, each child is subject to a random chance of mutation which begins at 1% and increases with each generation.

The IIRABM was optimized for 250 generations with a starting population size of 1024 candidate parameterizations. The IIRABM was implemented in C++ and the GA was implemented in Python 3; and simulations were performed on the Cori Cray XC40 Supercomputer at the National Energy Research Scientific Computing Center and at the Vermont Advanced Computing Center. Codes can be found at https://bitbcket.org/cockrell/iirabm_fullga/.

Results

A plot of cytokine ranges for 5 cytokines which existed in the clinical data set and were already present in the model at the start of this work (GCSF, TNF-α, IL-4, IL-10, and IFN-γ) is shown in

Figure 1. Ranges for the original model, described in (9, 10), are shown in black; ranges for the published data (20) are shown in red; and ranges for the optimized morel are shown in blue.

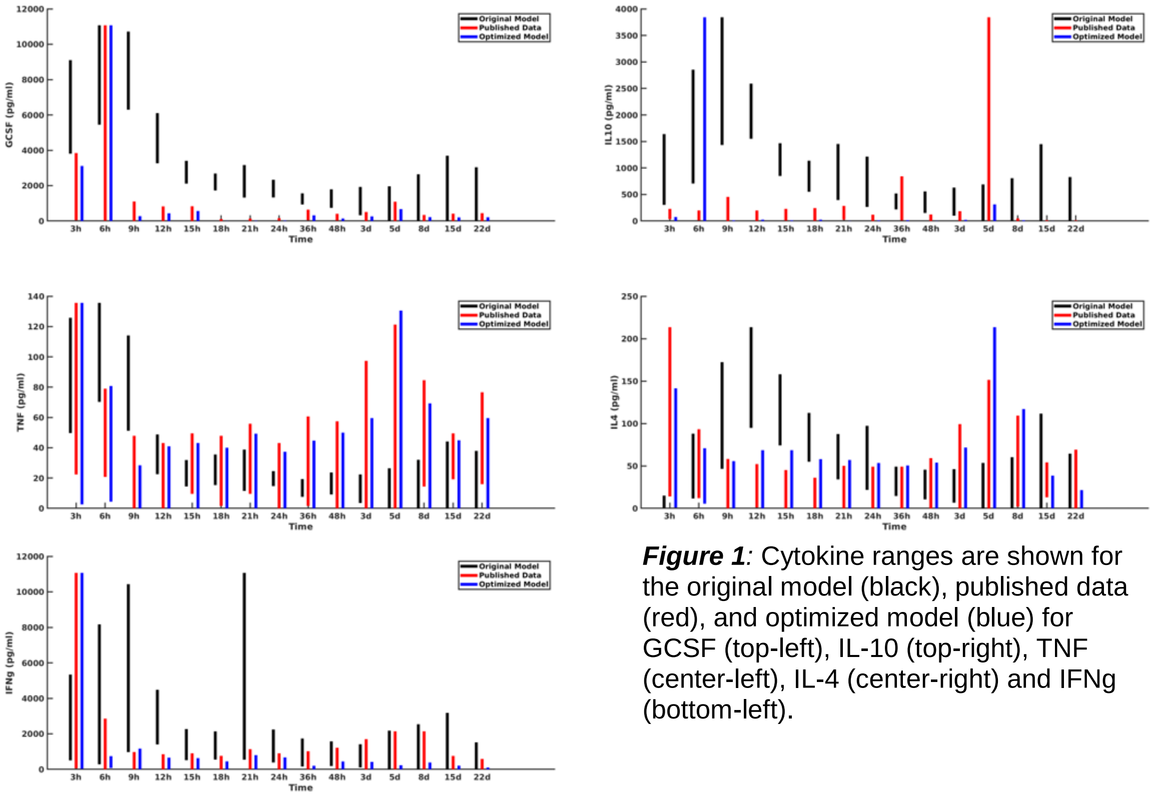

Figure 1*:* Cytokine ranges are shown for the original model (black), published data (red), and optimized model (blue) for GCSF (top-left), IL-10 (top-right), TNF (center-left), IL-4 (center-right) and IFNg (bottom-left).

We note that, while the model is optimized to closely match cytokine time courses for four out of the five cytokines used in the fitness function, IL-10 (Fig 1, top-right) does not match at well, with peaking occurring at 6 hours post-insult rather than 5 days post-insult, as was seen clinically. This discrepancy identifies a weakness in our model when it is being used to simulate burns, namely, that the cellular production of IL-10 is not well enough defined, in that its production is limited to activated macrophages and TH2 helper cells. Given that the IIRABM was developed to represent the innate immune response to traumatic injury, we consider this recalibration to burn injuries to be a success.

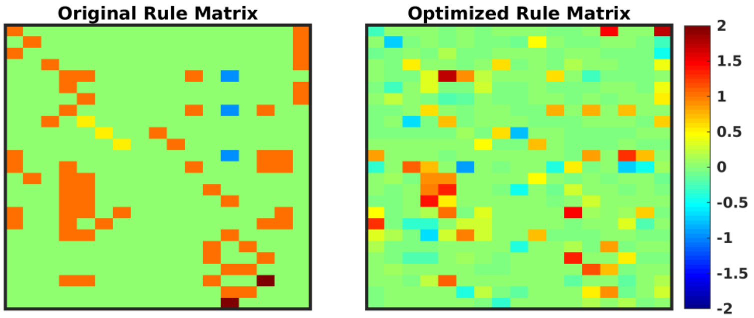

Figure 2: A heatmap of the original rule matrix is shown on the left and the optimized matrix is shown on the right. In these heatmaps, the light green represents a 0 or near-0 matrix element; the dark blue represents a negative matrix element; the red represents a positive matrix element.

We also posit that the nature of the IL-10 time series makes a poor fit more likely; the IL-10 time series spikes at t-5 days but is near zero everywhere else. A candidate MRM parameterization that minimizes IL-10 production over the entire time course would then contribute less to the overall fitness (in this case, we seek to minimize the fitness function) than a hypothetical parameterization that was 10%

off on TNF levels for every time step.

In Fig. 2, we compare the original rule matrix to the optimized rule matrix. Numerical values for both matrices can be found in the supplemental material. The optimized matrix has a much more connected structure, and is a dense matrix, as opposed to the sparse original rule matrix. There are not any matrix elements with a value of 0 in the optimized matrix, though there are many elements with comparatively small values. This structure is similar to what is seen in experimental bioinformatic studies; all of the cytokines in this network appear to be connected to each other, at least to a small degree, while a smaller number of strong connections (which could also be considered correlations) provide the majority of the influence on the system dynamics.

Discussion

The IIRABM rule set utilized in this work contained 432 free and continuous parameters, many of which had highly nonlinear or conditional effects on the model-generate cytokine trajectories and outcomes. Due to cytokine-specific properties, IL-10 was more challenging than the others when performing a multi-cytokine time series optimization. In future work, we will investigate the effects of both updating the cell types present in, and the structure of, the model and altering the fitness function. A simple fitness function alteration would be the addition of a constant multiplier in front of the IL-10 terms.

We note that by setting the fitness function to match the published data as closely as possible, we have neutralized the primary benefit of modeling, the minimal cost of adding another patient to the *in-silico* cohort. The true range of biologically plausible blood cytokine concentrations in undoubtedly larger than what is seen in a cohort of 20 patients. In order to obtain a more generalizable model, we propose two alternatives approaches to the above presented work: 1) that the fitness function should be configured to over-encompass the available data; and 2) that the fitness function incorporates the probability density function (pdf) which governs the experimental data. Incorporating the shape of the probability density function into the fitness function can be difficult purely as a matter of practicality – often the raw data for human cytokine levels isn't available, and only the absolute range can be extracted from published manuscripts, and it is also common to see a cohort size that is too small to definitively propose a single pdf which adequately describes the data.

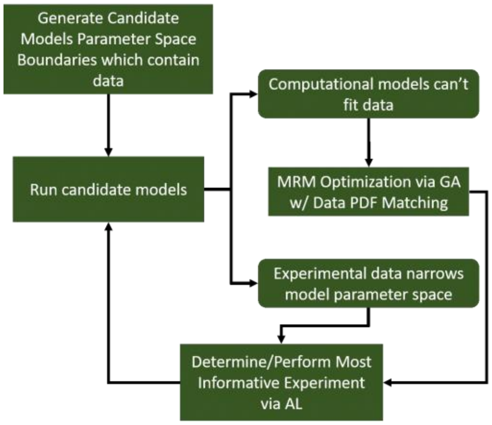
Generate Candidate Models Parameter Space Boundaries which contain data
Run candidate models
Computational models can't fit data
MRM Optimization via GA w/ Data PDF Matching
Experimental data narrows model parameter space
Determine/Perform Most Informative Experiment via AL

Figure 3: A diagram indicating a hybrid experimental/computational workflow for the automated calibration and validation of ABMs using the MRM scheme.

The result of GA model refinement and rule discovery is a model parameterization which, when instantiated dynamically, generates data that matches what is seen experimentally; in reality, this result represents a single step in an iterative cycle of model refinement and biological experimentation. At the conclusion of the GA run, there exists an ensemble of candidate model parameterizations which meet the fitness criterion to some fixed threshold. Many of the genes in each individual parameterization end up tightly constrained by the algorithm, while others have a larger range. These latter parameters are those about which the model is most uncertain. Active Learning is a sampling technique used in machine learning in which sampled data is chosen based on how much information it can apply to the machine learning model. A similar approach can be taken in this case. In order to most efficiently update and refine the computational model, experiments should be designed to query the model features that are most uncertain. This approach is illustrated in Fig. 3. In this way, GA can play an integral role in the iterative cycle of model refinement and experimentation necessary to construct a high-fidelity generalizable computational model.

1. An G. Dynamic knowledge representation using agent-based modeling: ontology instantiation and verification of conceptual models. Systems Biology: Springer; 2009. p. 445-68.
2. Bonabeau E. Agent-based modeling: methods and techniques for simulating human systems. Proceedings of the National Academy of Sciences of the United States of America. 2002;99 Suppl 3:7280-7.
3. Cockrell C, Christley S, An G. Investigation of Inflammation and Tissue Patterning in the Gut Using a Spatially Explicit General-Purpose Model of Enteric Tissue (SEGMEnT). PLoS computational biology. 2014;10(3):e1003507.
4. Cockrell RC, Christley S, Chang E, An G. Towards anatomic scale agent-based modeling with a massively parallel spatially explicit general-purpose model of enteric tissue (SEGMEnT_HPC). PLoS One. 2015;10(3):e0122192.
5. An G, Bartels J, Vodovotz Y. In Silico Augmentation of the Drug Development Pipeline: Examples from the study of Acute Inflammation. Drug Dev Res. 2011;72(2):187-200.
6. Cockrell C, Axelrod D. Optimization of Dose Schedules for Chemotherapy of Early Colon Cancer Determined by High Performance Computer Simulations. 2018.
7. Baldazzi V, Castiglione F, Bernaschi M. An enhanced agent based model of the immune system response. Cell Immunol. 2006;244(2):77-9.
8. Bailey AM, Thorne BC, Peirce SM. Multi-cell agent-based simulation of the microvasculature to study the dynamics of circulating inflammatory cell trafficking. Ann Biomed Eng. 2007;35(6):916-36.
9. Cockrell C, An G. Sepsis reconsidered: Identifying novel metrics for behavioral landscape characterization with a high-performance computing implementation of an agent-based model. J Theor Biol. 2017;430:157-68.
10. Cockrell RC, An G. Examining the controllability of sepsis using genetic algorithms on an agent-based model of systemic inflammation. PLoS computational biology. 2018;14(2):e1005876.
11. Ling Y, Mahadevan S. Quantitative model validation techniques: New insights. Reliability Engineering & System Safety. 2013;111:217-31.
12. Macal CM, editor Model verification and validation. Workshop on" Threat Anticipation: Social Science Methods and Models; 2005.
13. Calvez B, Hutzler G, editors. Parameter space exploration of agent-based models. International Conference on Knowledge-Based and Intelligent Information and Engineering Systems; 2005: Springer.
14. Abramson D, Bethwaite B, Enticott C, Garic S, Peachey T, editors. Parameter space exploration using scientific workflows. International Conference on Computational Science; 2009: Springer.
15. Carley KM. Validating computational models. Paper available at http://www.casos.cs.cmu. edu/publications/papers php. 1996.
16. Saltelli A, Annoni P. How to avoid a perfunctory sensitivity analysis. Environmental Modelling & Software. 2010;25(12):1508-17.
17. Ratto M, Pagano A, Young P. State dependent parameter metamodelling and sensitivity analysis. Computer Physics Communications. 2007;177(11):863-76.
18. Faix JD. Biomarkers of sepsis. Crit Rev Clin Lab Sci. 2013;50(1):23-36.
19. Arnold L. Random dynamical systems: Springer Science & Business Media; 2013.
20. Bergquist M, Hästbacka J, Glaumann C, Freden F, Huss F, Lipcsey M. The time-course of the inflammatory response to major burn injury and its relation to organ failure and outcome. Burns. 2019;45(2):354-63.
21. Haupt RL, Ellen Haupt S. Practical genetic algorithms. 2004.
22. Fonseca CM, Fleming PJ, editors. Genetic Algorithms for Multiobjective Optimization: FormulationDiscussion and Generalization. Icga; 1993: Citeseer.
23. Goldberg DE, Holland JH. Genetic algorithms and machine learning. Machine learning. 1988;3(2):95-9.
24. Haupt RL, Haupt SE. Practical genetic algorithms: John Wiley & Sons; 2004.