

RESEARCH ARTICLE

# Development of an Agent-Based Model (ABM) to Simulate the Immune System and Integration of a Regression Method to Estimate the Key ABM Parameters by Fitting the Experimental Data

Xuming Tong<sup>1</sup>, Jinghang Chen<sup>1\*</sup>, Hongyu Miao<sup>3</sup>, Tingting Li<sup>4\*</sup>, Le Zhang<sup>1,2\*</sup>

**1** College of Computer and Information Science, Southwest University, Chongqing, P. R. China, **2** Department of Biostatistics and Computational Biology, Center for Biodefense Immune Modeling, University of Rochester, New York, United States of America, **3** Department of Biostatistics, School of Public Health, University of Texas Houston, Houston, Texas, United States of America, **4** School of Mathematics and Statistics, Southwest University, Chongqing, China

\* [jinghang\\_chen@outlook.com](mailto:jinghang_chen@outlook.com) (JHC); [tinalee@swu.edu.cn](mailto:tinalee@swu.edu.cn) (TTL); [zhanglcq@swu.edu.cn](mailto:zhanglcq@swu.edu.cn) (LZ)



OPEN ACCESS

**Citation:** Tong X, Chen J, Miao H, Li T, Zhang L (2015) Development of an Agent-Based Model (ABM) to Simulate the Immune System and Integration of a Regression Method to Estimate the Key ABM Parameters by Fitting the Experimental Data. PLoS ONE 10(11): e0141295. doi:10.1371/journal.pone.0141295

**Editor:** Rongling Wu, Pennsylvania State University, UNITED STATES

**Received:** June 2, 2015

**Accepted:** October 7, 2015

**Published:** November 4, 2015

**Copyright:** © 2015 Tong et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All relevant data are within the paper and its Supporting Information files.

**Funding:** This work was supported by the following sources of funding: 1. Natural Science Foundation of China No. 61372138 (<http://www.nsf.gov.cn/>); 2. USA National Institutes of Health grants U01 CA166886-01, P30AI078498, and HHSN272201000055C (<http://grants.nih.gov/grants/oeer.htm>); 3. Chongqing excellent youth award (<http://www.ctin.ac.cn/>); and 4. Chinese Recruitment

## Abstract

Agent-based models (ABM) and differential equations (DE) are two commonly used methods for immune system simulation. However, it is difficult for ABM to estimate key parameters of the model by incorporating experimental data, whereas the differential equation model is incapable of describing the complicated immune system in detail. To overcome these problems, we developed an integrated ABM regression model (IABMR). It can combine the advantages of ABM and DE by employing ABM to mimic the multi-scale immune system with various phenotypes and types of cells as well as using the input and output of ABM to build up the Loess regression for key parameter estimation. Next, we employed the greedy algorithm to estimate the key parameters of the ABM with respect to the same experimental data set and used ABM to describe a 3D immune system similar to previous studies that employed the DE model. These results indicate that IABMR not only has the potential to simulate the immune system at various scales, phenotypes and cell types, but can also accurately infer the key parameters like DE model. Therefore, this study innovatively developed a complex system development mechanism that could simulate the complicated immune system in detail like ABM and validate the reliability and efficiency of model like DE by fitting the experimental data.

Program of Global Youth Experts (<http://www.1000plan.org/>).

**Competing Interests:** The authors have declared that no competing interests exist.

## Introduction

Currently, system biologists employ agent-based models (ABM) [1–5] and differential equation models (DE) [6–9] to simulate the immune system. Detailed definitions of ABM and DE are illustrated in the [S1 File](#).

Recently, researchers did develop several ABMs for the immune system simulation. For example, The Basic Immune Simulator (BIS) [10] is an agent-based model (ABM) that can be used to study the interactions between cells of the innate and adaptive immune systems. The BIS demonstrated that the degree of the initial innate response was a crucial determinant for an appropriate adaptive response [10]. Also, the ImmunoGrid project [11] is to develop a natural-scale model of the human immune system using an ABM, that can reflect both the diversity and the relative proportions of the molecules and cells. This model will be of great value for specific applications in the field of immunology [11].

ABM has several significant advantages. First, its natural representational formalism can be employed to denote a cell's biological properties and behavior in detail [1]. Second, its flexible features can be employed to reflect the real complex dynamic environment [12]. However, it is difficult for ABM to incorporate experimental data, because ABM describes the system at the level of its constituent units but not at the top level [13].

DE is broadly employed to approximate experimental data and predict the progression of the immune system. For example, researchers have applied it to the case of influenza A virus (IAV) infection. Miao et al., [14] developed a differential equation model to describe the dynamic interactions among the components (i.e., epithelial cells, virus, CD8 CTLs, and antibody) in the lung. The model was used to quantify the immune responses and to estimate the key parameters in primary infection. Not limited to IAV infection, DE can also be widely used for other virus infections, such as HIV in the study of Miao et al. [9]. The researchers developed statistical estimation, model selection, and multi-model averaging methods for in vitro HIV viral fitness experiments using a set of nonlinear ordinary differential equations and addressed the parameter identifiability of the model [9].

The DE has been the focus of a great deal of attention due to its great potential as a new optimization technique to solve complex nonlinear problems and widespread use in various areas [15]. Compared to ABM, DE can be easily employed to solve the optimization problem by estimating a few control parameters [15]. However, it has difficulty describing the details of biological systems because DE falls short in constructing a biological model to a sufficient degree, especially when faced with the simulation of complex phenomena.

To integrate the advantages of these two commonly used models, we developed an integrated ABM regression model (IABMR) and employed the IAV data set [14] to evaluate its efficiency and accuracy. IABMR employed ABM to denote each cell as an agent with three phenotypes (i.e., quiescence, proliferation and apoptosis). Then, it employed Loess regression to build a Loess model based on the input and output of ABM. The model's key parameters were optimized using the particle swarm optimization algorithm (PSO) [16–21]. The concept of PSO is illustrated in the [S1 File](#).

Next, we employed the classical greedy algorithm [22–24] to optimize the ABM parameter and compare the efficiency of ABM with the greedy algorithm and IABMR. The results demonstrated that IABMR not only described the immune response at the cellular level using various cells' phenotypes and possessed great potential for investigating interactions and special information for the cells but also overcame the limitations of ABM in parameter estimation.

## Methods

### 2.1. Using ABM to describe the immune system

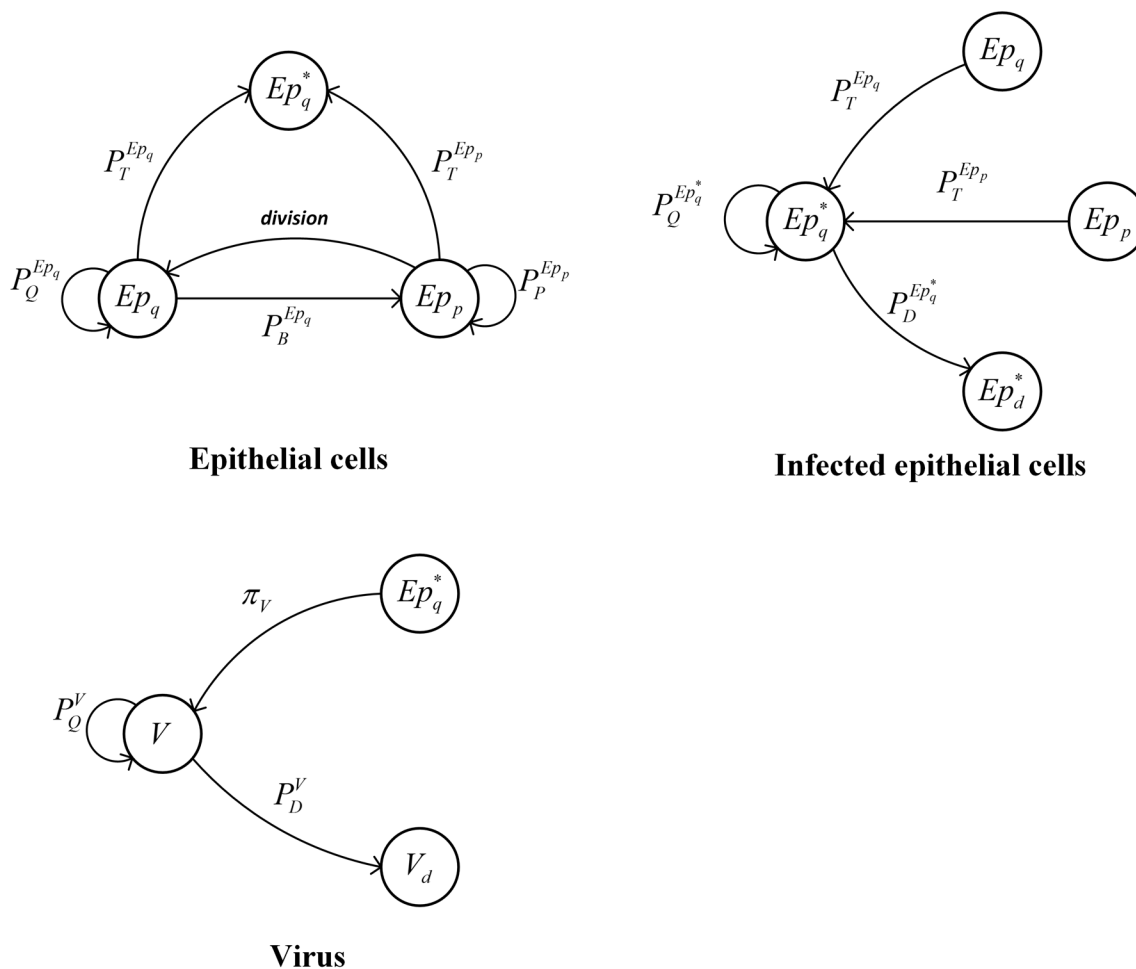
To describe the dynamic interactions among the components (i.e., epithelial cells, infected epithelial cells and virus) in the lung, Fig 1 is used to quantify immune responses in primary infections.

An epithelial cell in a quiescent state  $Ep_q$  can be transited to three other states. Two of these states belong to the  $Ep$  cell, where  $P_B^{Ep_q}$  and  $P_Q^{Ep_q}$  are the probabilities for  $Ep_q$  to change its state.  $Ep_q$  and  $Ep_p$  are two states of the  $Ep$  cell. The  $Ep_q$  cell can also be differentiated into another type of the cell ( $Ep^*$ ) with a probability  $P_T^{Ep_q}$ . Once the  $Ep_q$  state transits to the  $Ep_p$  state with a probability of  $P_B^{Ep_q}$ , it will have  $P_P^{Ep_p}$  and  $P_T^{Ep_p}$  probabilities to become  $Ep_p$  and  $Ep_q^*$ , respectively.

With respect to the above state transition diagram (Fig 1 (Epithelial cells)), the state transition equations for epithelial cells are developed as follows.

$$Ep_{q(t)} = Ep_{q(t-1)}P_Q^{Ep_q} + 2Ndiv_t^{Ep} - Ep_{q(t-1)}P_T^{Ep_q}V_{t-1} - Ep_{q(t-1)}P_B^{Ep_q} \tag{1.1}$$

$$Ep_{p(t)} = (Ep_{p(t-1)} - Ndiv_t^{Ep})P_P^{Ep_p} + Ep_{q(t-1)}P_B^{Ep_q} - Ep_{p(t-1)}P_T^{Ep_p}V_{t-1} \tag{1.2}$$



**Fig 1. State transition diagrams of epithelial cells, infected epithelial cells and virus.**

doi:10.1371/journal.pone.0141295.g001

Here,  $V$  represents the infective viral titer and  $Ndiv_t^{Ep}$  is used to represent the number of cells which will divided into two cells. The case of an infected epithelial cell is shown in [Fig 1 \(Infected epithelial cells\)](#).

The  $Ep_q^*$  state can transit to itself and  $Ep_d^*$  with the probability  $P_Q^{Ep_q^*}$  and  $P_D^{Ep_q^*}$ , respectively. The transition equations are described as the following equations.

$$Ep_{q(t)}^* = Ep_{q(t-1)}^* P_Q^{Ep_q^*} + Ep_{q(t-1)} P_T^{Ep_q} + Ep_{p(t-1)} P_T^{Ep_p} - Ep_{q(t-1)}^* P_D^{Ep_q^*} \tag{2.1}$$

$$Ep_{d(t)}^* = Ep_{q(t-1)}^* P_D^{Ep_q^*} + Ep_{d(t-1)}^* \tag{2.2}$$

Different from the epithelial cell and infected epithelial cell, the virus is too small to be described as a discrete variable. In [Fig 1 \(Virus\)](#), the virus is described as a continuous variable with  $P_D^V$  percentage of dying ( $V_d$  state) and  $P_Q^V$  percentage of living. Here, we set  $P_D^V + P_Q^V = 1$ . Additionally, the virus can be produced by  $Ep_q^*$  with respect to the rate of  $\pi_v$ .

The case of the virus can be described using the following equations.

$$V_t = V_{t-1} P_Q^V + Ep_{q(t-1)}^* \pi_v - V_{t-1} P_D^V \tag{3.1}$$

$$V_{d(t)} = V_{t-1} P_D^V + V_{d(t-1)} \tag{3.2}$$

To simulate the process of cellular immunity among the epithelial cells, virus and infected epithelial cells, an agent based model (ABM) is developed based on the diagrams and equations provided above. The parameters listed in [Table 1](#) agree with the following rules.

$$P_T^{Ep_q} + P_B^{Ep_q} + P_Q^{Ep_q} = 1 \tag{4.1}$$

$$P_T^{Ep_p} + P_p^{Ep_p} = 1 \tag{4.2}$$

$$P_T^{Ep_q} = P_T^{Ep_p} \tag{4.3}$$

$$P_Q^{Ep_q^*} + P_D^{Ep_q^*} = 1 \tag{4.4}$$

$$P_Q^V + P_D^V = 1 \tag{4.5}$$

## 2.2. Parameter Estimation

To estimate the parameters in this study, parameter vector space ( $H$ ) is generated by the Sparse Grid method [25], which consists of a set of parameter vectors; each vector has 4 dimensions. The Sparse Grid method always chooses the most important points in the high dimension space to approximate the complicated surface [26–28].

In what follows, the input parameter of ABM is denoted by a four-dimensional vector  $\theta$ , where the components  $\theta_k$ ,  $k = 1,2,3,4$  represents  $(P_B^{Ep_q}, P_T^{Ep_q}, P_D^{Ep_q^*}, P_D^V)$  respectively. Reported by the previous research [14], the input data  $\theta$  are estimated as  $(6.2 \times 10^{-9}, 2.42 \times 10^{-7}, 5.98 \times 10^{-2}, 4.23 \times 10^{-1})$ , which we call as the initial parameter  $\theta_0$ . In this study, we set the input parameter of ABM in the region  $(0, 2\theta_0) = \{(\theta_1, \theta_2, \theta_3, \theta_4) \in R^4, 0 \leq \theta_k \leq 2\theta_{0k}, k = 1,2,3,4\}$ . However, according to the rules of the Sparse Grid, each component of parameter vector  $h \in H$  is between 0 and 1. Therefore, we

**Table 1. Parameters and variables definitions for agent based model.**

Parameter	Definition	Value
$P_T^{Ep_q}$	Infection rate of $Ep_q$ ( $hour^{-1}$ )	$2.42 \times 10^{-7}$
$P_T^{Ep_p}$	Infection rate of $Ep_p$ ( $hour^{-1}$ )	$2.42 \times 10^{-7}$
$P_B^{Ep_q}$	Proliferation rate of $Ep_q$ ( $hour^{-1}$ )	$6.2 \times 10^{-9}$
$P_Q^{Ep_q}$	Quiescence rate of $Ep_q$ ( $hour^{-1}$ )	$9.999997518 \times 10^{-1}$
$P_P^{Ep_p}$	Probability value for $Ep_p$ to stain resting ( $hour^{-1}$ )	$9.999997518 \times 10^{-1}$
$P_Q^{Ep_q^*}$	Probability value for $Ep_q^*$ to stain resting ( $hour^{-1}$ )	$9.402 \times 10^{-1}$
$P_D^{Ep_q^*}$	Death rate of $Ep_q^*$ ( $hour^{-1}$ )	$5.98 \times 10^{-2}$
$\pi_v$	Virus productivity of $Ep_q^*$ ( $hour^{-1}$ )	$1.0 \times 10^1$
$P_D^V$	Death rate of $V$ ( $hour^{-1}$ )	$4.23 \times 10^{-1}$
$P_Q^V$	Survival rate of $V$ ( $hour^{-1}$ )	$5.77 \times 10^{-1}$

doi:10.1371/journal.pone.0141295.t001

need to map the parameter vector space  $H$  generated by Sparse Grid into the region  $(0, 2\theta_0)$ . The mapping function is:

$$\theta_1 = (b - a)h + a \tag{5}$$

Where  $h$  is a parameter vector in the space  $H$ ,  $a = 0$ ,  $b = 2\theta_0$ .

$\theta_1$  is employed as the input parameters for the ABM to generate  $L$  sets of output data ( $G_1$ ), which represents the number of cells in 5 days. To generate randomness for ABM, we performed  $Lr$  replicates for each set of  $\theta_1$ . Next,  $\theta_1$  and  $G_1$  are employed to develop a Loess regression [29–32] mode  $M_0$ .

In our model  $M_0$ , the Loess regression is described in Eq 6.

$$\chi^2 = \sum_i w(\theta_{1i} - \mathbf{x}; g)(\alpha + \beta(\theta_{1i} - \mathbf{x}) - G_{1i})^2 \tag{6}$$

Here,  $w$  is a weighting function and  $\theta_{1i}$  is an input parameter of ABM, where  $i$  denotes the  $i$ -th sampling point in the parameter vector space.  $\theta_{1i}$  represents the points in the neighborhood of  $\mathbf{x}$  to be weighted by  $w$  depending on the distance to  $\mathbf{x}$ .  $g$  is a key parameter in the procedure called the "bandwidth" or "smoothing parameter" that determines how much of the data is used to fit each local polynomial.  $G_{1i}$  is the output data value of ABM corresponding to the input data  $\theta_{1i}$ .  $\alpha$  and  $\beta$  are two coefficients of the least squares method [33] that is employed to approximate their value by minimizing the value of  $\chi^2$  in Eq 6.

Next, the particle swarm optimization algorithm (PSO) [16] is employed to locate the optimal parameter by fitting the real experimental data. PSO [17–21] is illustrated in the S1 File in detail, and its key equations are described by Eqs 7.1 and 7.2.

$$v_{i(t+1)} = wv_{i(t)} + c_1 \cdot rand() \cdot (p_{i(t)} - x_{i(t)}) + c_2 \cdot Rand() \cdot (p_{g(t)} - x_{i(t)}) \tag{7.1}$$

$$x_{i(t+1)} = x_i + v_{i(t+1)} \tag{7.2}$$

First, let  $S$  be the number of particles in the swarm. Then initialize the particle's position with a uniformly distributed random vector  $x_i \sim U(lb, ub)$ , where  $lb$  and  $ub$  are the lower and upper boundaries of the search-space, here  $(lb, ub) = (0, 2\theta_0)$ . Obviously,  $x_i$  can be considered as the input parameter. The particle's initial velocity is:  $v_i \sim U(-|ub-lb|, |ub-lb|)$ . Here,  $w$  is a weight function used to maintain the inertia force of each particle. Let  $p_i$  be the best known position of particle  $i$  and let  $p_g$  be the best known position of the entire swarm. Then, Eq 8 is

employed as the object function for the parameter estimation.

$$f_{obj} = \sum_{j=1}^m \sum_{i=1}^n (y_i - V_1)^2 \tag{8}$$

Here,  $m$  is the time point, and  $n$  is the replicates at each time point,  $V_1$  is the real experimental data in five days.  $y_i$  is the predictive value from the Loess model based on input value  $x_i$ .

By using the PSO algorithm and Loess model, we can minimize the object function  $f_{obj}$  to locate the local optimal parameter  $\theta^*$  in the region  $(\mathbf{0}, 2\theta_0)$ .

Next, we reemployed the mapping function (Eq 5) to map parameter vector space  $H$  on region  $(\mathbf{0}, 2\theta^*)$  to generate  $L$  sets of input parameters  $\theta_2$  and  $n$  replicates for each set of  $\theta_2$ . These data will be employed as input parameters in the ABM; then, we can obtain  $G_2$  output data with  $m$  time points. Next,  $\theta^*$  will be employed as the input data of ABM to generate the simulated experimental data set  $V_2$  with  $n$  replicates, which will replace  $V_1$  by adding random noise.

The normal distribution method (Eqs 9.1 and 9.2) [34] is used to add noise for each replicate of the  $V_2$  data set and develop the simulated experimental data set  $V_2^*$ .

$$V_\Delta \sim N(0, \alpha_i) \quad i = 1, 2, 3. \tag{9.1}$$

$$V_2^* = V_2 + V_\Delta \tag{9.2}$$

$N(0, \alpha_i)$  denotes a normal distribution with mean 0 and standard deviation  $\alpha_i$ .

Next, a new Loess regression model  $M_1$  is built based on  $\theta_2$  and  $G_2$  in a process similar to  $M_0$ . We used PSO [35] to explore the optimal local parameter  $Est\theta_i$  by fitting the simulated experimental data  $V_2^*$ . Finally, we can compute average relative error (ARE) [9] for each  $Est\theta_i$  using Eq 10.

$$ARE = \sum_{i=1}^M \frac{|Est\theta_i - \theta^*|}{M \times |\theta^*|} \times 100\% \tag{10}$$

Here,  $M$  is the total number of ABM simulation runs for each sample. This parameter estimation process is illustrated in Fig 2.

## Results

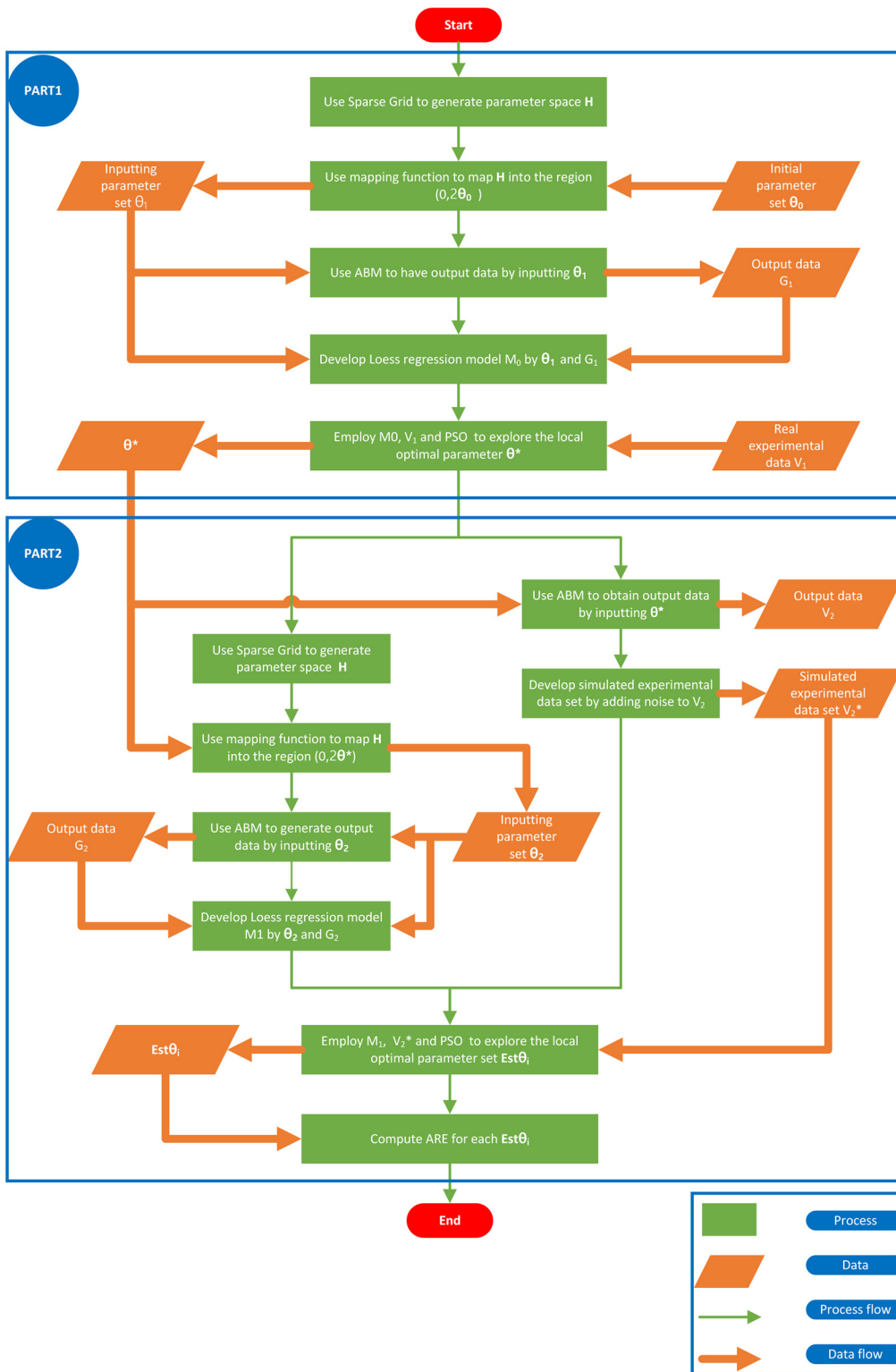
The IABMR model is developed using C++ and R program language and works in the Linux environment.

### 3.1. Primary data for model fitting

We used real experimental data  $V_1$ [14] from infection of mice with the H3N2 influenza virus A/X31 strain to fit the model. This study employs data from the initial preadaptive phase constituting 0 to 5 days post-infection. The real experimental data contains 6 samples and each sample has 13 time points. The detailed experimental data information is listed in Table 2. The initial key parameters of ABM are also from the literature [14].

### 3.2. Obtain the sampling data using Sparse Grid function

We employed the “createIntegrationGrid” function of the R “SparseGrid” package to create three sampling data sets in the region  $(\mathbf{0}, \mathbf{1})$  (sample size: 41, 137 and 385) (listed in S1–S3 Tables). Then, these sampling data are mapped to the input parameters sets of ABM ( $\theta_1$ ) by Eq 5. The values of  $\theta_1$  are listed in S4–S6 Tables.



**Fig 2. Parameter estimation process.**

doi:10.1371/journal.pone.0141295.g002

Table 2. Real experimental data between 0 to 5 days.

samples	Time points (day <sup>-1</sup> )												
	0	0.125	0.25	0.5	1	1.5	2	2.5	3	3.5	4	4.5	5
1	4.25	2.5	3.5	4.25	5.5	6.5	6.33	6.75	6.5	6.5	6.5	7	6.33
2	3.75	2.5	4.75	3.25	6.75	6.75	7.5	3.5	7.33	7.25	6.25	6.5	5.5
3	4.25	3.5	4.75	5.25	6.5	7.75	7.75	7.5	7.33	7.25	6.5	6.25	5.75
4	3.75	3.5	4.13	5.75	7.25	NA	7.25	6.5	6.25	5.5	NA	NA	NA
5	4.55	2.75	2.5	5.75	NA	NA	NA	7.5	6.75	6.5	NA	NA	NA
6	4.25	NA	4.75	5.5	NA	NA	NA	NA	7.25	5.75	NA	NA	NA

doi:10.1371/journal.pone.0141295.t002

### 3.3. Estimate the parameter of ABM by fitting the real experimental data

To obtain randomness, we run data sample 41, 137 and 385 with 9,9 and 6 times. And then, we denote them as model 41×9, 137×9 and 385×6, respectively. The output data set  $G_1$  (S7–S9 Tables) of ABM is obtained by inputting  $\theta_1$ . Eqs 7.1 and 7.2 is employed to explore the local optimum parameter  $\theta^*$  for each sampling data set listed in Table 3.

### 3.4. Generate the simulated experimental data by ABM

We can obtain an output of ABM  $V_2$  by inputting  $\theta^*$ . The simulated experimental data  $V_2^*$  is developed from  $V_2$  by Eq 9 by adding three levels of noise ( $\alpha_i$ ), such as  $\sqrt{0.75}$ ,  $\sqrt{1.50}$  and  $\sqrt{3.00}$  regarding to our previous study [36]. Part of the simulated experimental data is listed in S10–S12 Tables.

### 3.5. Average relative error computing

After fitting the model to the simulated experimental data using Eqs 7.1 and 7.2, we obtain the local optimal parameter  $Est\theta_i$ . Then, Eq 10 is employed to compute the average relative error for each set of simulated experimental data. Here, we set the total number of ABM simulation runs as  $M = 100$  and the three sample sizes as 5×3 (5 is time points ( $m$ ), 3 is the replicates ( $n$ )), 10×6 and 15×9. The values of ARE for each sample size are listed in Tables 4–6.

### 3.6. Evaluate the accuracy and efficiency of the IABMR model

To evaluate the accuracy and efficiency of the IABMR model in parameter estimation, we employed the greedy algorithm [22,37] with ABM to estimate the parameters. Fig 3 compares their residual errors (RSS). Here, RSS1 is the residual errors of the greedy algorithm as well as RSS2, RSS3 and RSS4 are the residual errors of the three sampling data sets from IABMR (model 41×9,137×9 and 385×6).

Table 3. Optimum parameter for each sampling model.

Sampling model	optimum parameter ( $\theta^*$ ) for each model			
	$P_B^{EPq}$	$P_T^{EPq}$	$P_D^{EPq}$	$P'_D$
41×9	0.00000006200000	0.00000005454770	0.01372567	0.7506544
137×9	0.00000006170504	0.00000003664683	0.01600081	0.6807255
385×6	0.00000006146936	0.00000004597134	0.01651409	0.6676635

doi:10.1371/journal.pone.0141295.t003



**Table 4. The summary table of ARE values for model 41x9.**

Sample size (m×n)		$ARE = \sum_{i=1}^M \frac{ Est\theta_i - \theta^* }{M \times  \theta^* } \times 100\%$			
m:time point n:replicates	Noise level	$P_B^{EPq}$	$P_T^{EPq}$	$P_D^{EPq}$	$P_D^V$
5x3	$\sqrt{0.75}$	0.396374135606522	0.443644533769457	0.056352115904734	0.443648916301695
10x6	$\sqrt{0.75}$	0.397479215621607	0.288728219873243	0.056352115904734	0.288729750509077
15x9	$\sqrt{0.75}$	0.396050197041251	0.288728219873243	0.056352115904734	0.288729750509077
5x3	$\sqrt{1.50}$	0.401597828194687	0.133811905977029	0.056352115904734	0.443648916301695
10x6	$\sqrt{1.50}$	0.355114873293543	0.211270062925136	0.056352115904734	0.443648916301695
15x9	$\sqrt{1.50}$	0.303420318533482	0.288728219873243	0.056352115904734	0.443648916301695
5x3	$\sqrt{3.00}$	0.056351812916928	0.095082827502976	0.056352115904734	0.169609252886410
10x6	$\sqrt{3.00}$	0.056351812916928	0.133811905977029	0.056352115904734	0.120094487389536
15x9	$\sqrt{3.00}$	0.056351812916928	0.095082827502976	0.056352115904734	0.155694584660783

doi:10.1371/journal.pone.0141295.t004

**Table 5. The summary table of ARE values for model 137x9.**

Sample size (m×n)		$ARE = \sum_{i=1}^M \frac{ Est\theta_i - \theta^* }{M \times  \theta^* } \times 100\%$			
m:time point n:replicates	Noise level	$P_B^{EPq}$	$P_T^{EPq}$	$P_D^{EPq}$	$P_D^V$
5x3	$\sqrt{0.75}$	0.643522575254619	0.930561319026966	0.399535317487816	0.930566159038856
10x6	$\sqrt{0.75}$	0.544230567172763	0.356479937741299	0.087188402282124	0.912747140886330
15x9	$\sqrt{0.75}$	0.370431446243038	0.579838310802485	0.069451181484305	0.686363123221724
5x3	$\sqrt{1.50}$	0.920087152935404	0.777707452418573	0.319657912364586	0.930566159038856
10x6	$\sqrt{1.50}$	0.678690815711458	0.643520628384133	0.165464682512184	0.930566159038856
15x9	$\sqrt{1.50}$	0.356478932929542	0.513177041198269	0.069451181484305	0.376377320727785
5x3	$\sqrt{3.00}$	0.643522575254619	0.930561319026966	0.494169438477352	0.930566159038856
10x6	$\sqrt{3.00}$	0.497055974244236	0.643520628384133	0.107579631811147	0.922833014750482
15x9	$\sqrt{3.00}$	0.426123861132435	0.235982026649789	0.147703266116924	0.643522053012952

doi:10.1371/journal.pone.0141295.t005

**Table 6. The summary table of ARE values for model 385x6.**

Sample size (m×n)		$ARE = \sum_{i=1}^M \frac{ Est\theta_i - \theta^* }{M \times  \theta^* } \times 100\%$			
m:time point n:replicates	Noise level	$P_B^{EPq}$	$P_T^{EPq}$	$P_D^{EPq}$	$P_D^V$
5x3	$\sqrt{0.75}$	0.558195074749436	0.377709938409453	0.734336363674898	0.467353749306350
10x6	$\sqrt{0.75}$	0.550373063913468	0.281480374511598	0.646693956494121	0.428673366149265
15x9	$\sqrt{0.75}$	0.317992736543865	0.259980152851755	0.474146550006691	0.215504157408635
5x3	$\sqrt{1.50}$	0.640824859734996	0.421591800456545	0.820570179767702	0.578291010366749
10x6	$\sqrt{1.50}$	0.563205571035716	0.329446998934554	0.686081473456909	0.497756369788074
15x9	$\sqrt{1.50}$	0.545574901056396	0.296157823548324	0.551033947374636	0.382253155968538
5x3	$\sqrt{3.00}$	0.734766947305129	0.497921805194280	0.821690326260787	0.587863332352300
10x6	$\sqrt{3.00}$	0.606827857000626	0.425248426519653	0.762389535239302	0.543190334652111
15x9	$\sqrt{3.00}$	0.568049903236344	0.415941323441953	0.563435357322141	0.407836342708565

doi:10.1371/journal.pone.0141295.t006

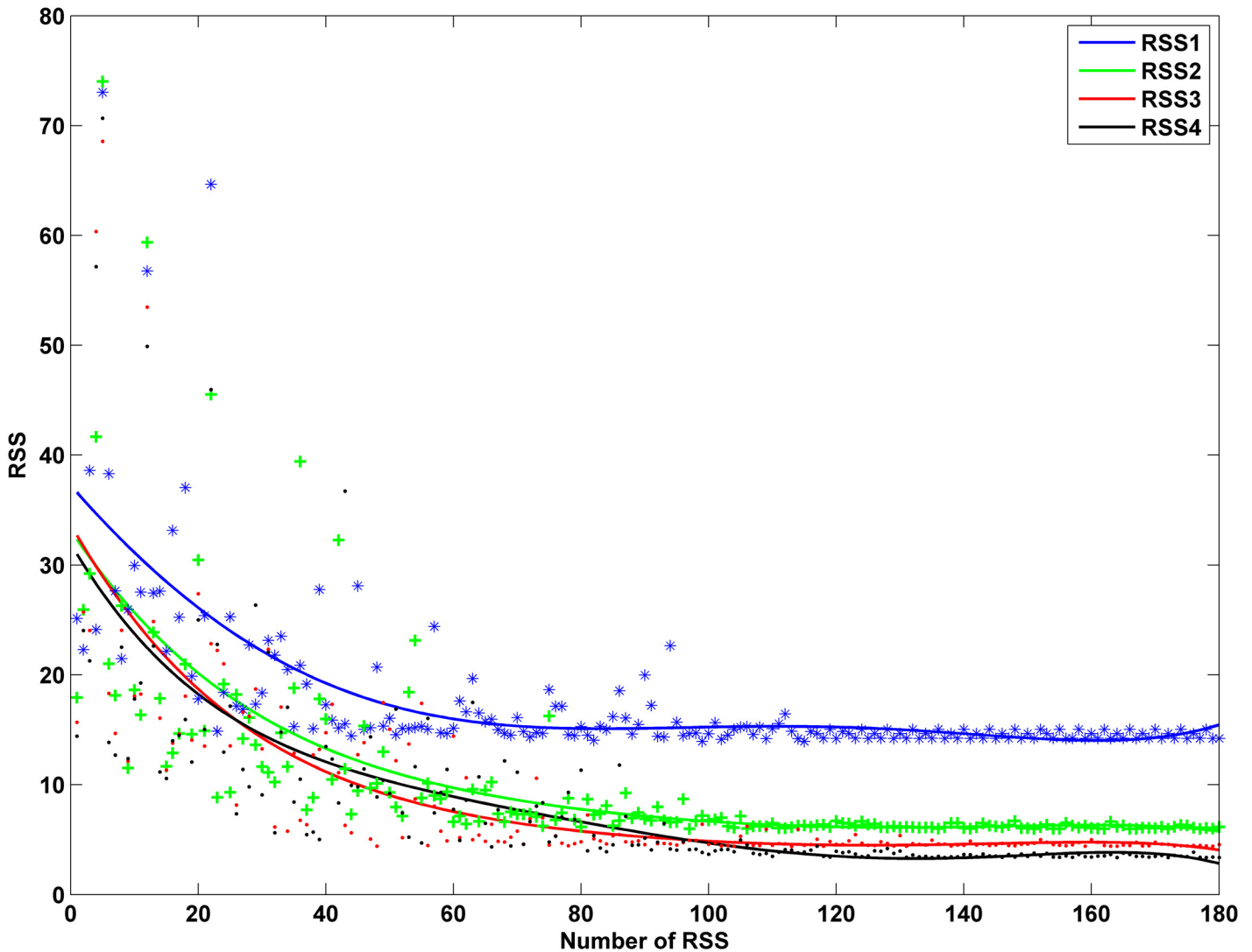


Fig 3. Comparison among RSS1, RSS2, RSS3 and RSS4.

doi:10.1371/journal.pone.0141295.g003

### 3.7. Using IABMR to approximate primary data

Fig 4 illustrates that IABMR can approximate the primary data with a similar effect as the ODE model [14].

### Discussion

In this work, we developed an agent-based model (ABM) to simulate influenza A virus (IAV) infection and integrated the ABM with Loess regression to develop an integrated ABM regression model (IABMR). This model can be employed to locate the key ABM parameter by fitting the real experimental data.

By inheriting the advantages of ABM, IABMR is capable of mimicking the biological system in detail. Here, IABMR not only showed quantitative changes in the system but also simulated the phenotypic switch for each cell type. Compared to the previous well-developed ODE model [14], it was possible to describe a multi-scale biological system in a very complicated external

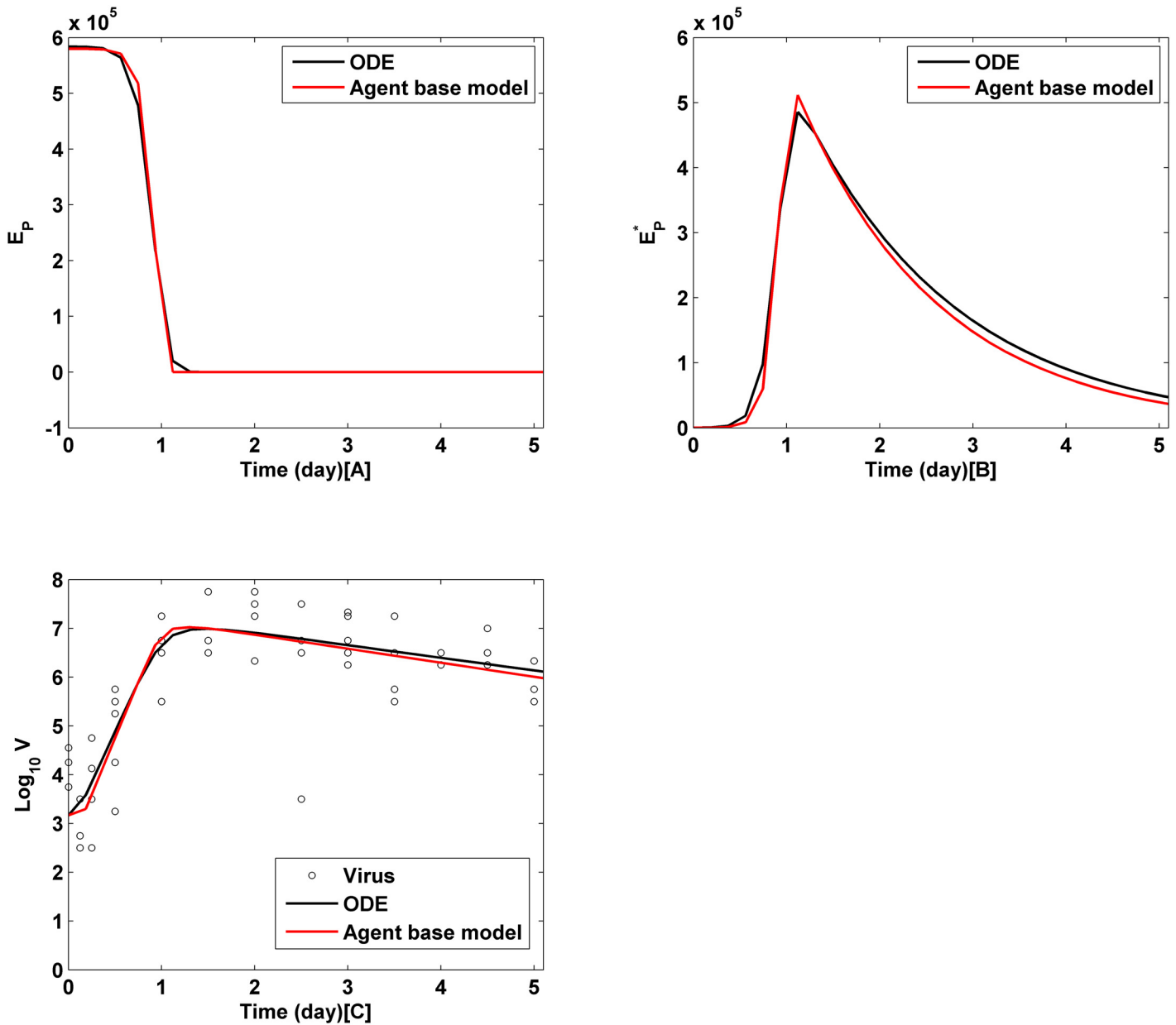


Fig 4. Comparison between IABMR and ODE.

doi:10.1371/journal.pone.0141295.g004

environment. IABMR Integrated with Loess regression [29] can employ classical numerical optimization methods such as the genetic algorithm [38,39] to estimate key parameter of the model, which is much faster than the greedy algorithm [22–24] used by ABM. These two theoretical advantages made IABMR an attractive application to simulate biological systems, compared to the ODE and ABM.

The average relative error (ARE) is commonly employed to evaluate the capacity of parameter estimation for statistical models. The smaller the ARE, the better the model’s performance. Tables 4–6 showed the ARE values of four key probabilities of the IABMR under the control of the following two aspects: the number of time points collected from the preadaptive phase and the level of noise added to the simulated experimental data.

[Table 6](#) showed two trends of ARE under different noise levels and numbers of time points. First, the ARE values decrease when the number of time points increases from 5 to 15 at the same noise level. For example, the ARE value of  $P_B^{Epq}$  has the order  $P_B^{Epq}(5 \times 3) > P_B^{Epq}(10 \times 6) > P_B^{Epq}(15 \times 9)$  under noise  $\sqrt{0.75}$ , which indicates more time points and replicates can obtain better parameter estimation accuracy.

Second, the ARE values increase when the noise level increases from  $\sqrt{0.75}$  to  $\sqrt{3.00}$  under the same number of time points, which demonstrates that the parameter estimation accuracy is higher with a smaller noise level. For instance, in the case of sample  $5 \times 3$  ([Table 6](#)), the ARE value of  $P_B^{Epq}$  has the order:

$P_B^{Epq}(\sqrt{0.75}) < P_B^{Epq}(\sqrt{1.50}) < P_B^{Epq}(\sqrt{3.00})$  The additional three probabilities ( $P_T^{Epq}$ ,  $P_D^{Epq}$  and  $P_D^v$ ) in the parameter have similar trends to  $P_B^{Epq}$  ([Table 6](#)).

[Fig 3](#) compared the accuracy and parameter estimation speed between the IABMR and ABM models. IABMR is much faster than ABM in terms of locating key parameter. For example, it takes at least 54,600 runs for ABM with the greedy algorithm to make the RSS converge, but only 2310 runs for IABMR with the largest size of parameter space to make the RSS converge. Additionally, the size of the parameter vector space has high impact on the parameter estimation accuracy. The larger the size, the more accurate the estimated results. As described by the [Fig 3](#), model  $41 \times 9$  has the greatest RSS and model  $385 \times 6$  has the least RSS. Meanwhile, the trends of the ARE values in [Tables 4](#) and [5](#) are not as perfect as in [Table 6](#). Lastly, [Fig 4](#) demonstrated that the IABMR simulation results had high similarity like the ODE to approximate the real experiential data, which validated the efficiency and accuracy of the IABMR.

In conclusion, this study developed an IABMR method to simulate detailed biological systems and locate their key parameter using classical numerical optimization methods. By integrating the advantages of both the ABM and ODE modes, it not only described the complicated microenvironment of the biological system and the cell's behavior in multiple scales in detail, but also easily to incorporate real experimental data. To evaluate the efficiency and accuracy of IABMR, we employed primary influenza infection data as the case study to exhibit the advantages of the IABMR. The validation results demonstrated that IABMR could mimic the immune system on multiple levels similar to ABM and approximate real experimental data similar to ODE with a reasonable parameter estimation cost.

## Supporting Information

**S1 File. The introduction of ABM, DE and PSO.**

(PDF)

**S1 Table. Sample size 41 generated by Sparse Grid.**

(PDF)

**S2 Table. Sample size 137 generated by Sparse Grid.**

(PDF)

**S3 Table. Sample size 385 generated by Sparse Grid.**

(PDF)

**S4 Table. Input data of ABM mapped from sample size 41.**

(PDF)

**S5 Table. Input data of ABM mapped from sample size 137.**

(PDF)

**S6 Table. Input data of ABM mapped from sample size 385.**

(PDF)

**S7 Table. Output data sets of model 41×9.**

(PDF)

**S8 Table. Output data sets of model 137×9.**

(PDF)

**S9 Table. Output data sets of model 385×6.**

(PDF)

**S10 Table. Sample size 5×3 with noise  $\sqrt{0.75}$ .**

(PDF)

**S11 Table. Sample size 5×3 with noise  $\sqrt{1.50}$ .**

(PDF)

**S12 Table. Sample size 5×3 with noise  $\sqrt{3.00}$ .**

(PDF)

## Author Contributions

Conceived and designed the experiments: XMT JHC HYM TTL LZ. Performed the experiments: XMT JHC HYM TTL LZ. Analyzed the data: XMT JHC HYM TTL LZ. Contributed reagents/materials/analysis tools: XMT JHC HYM TTL LZ. Wrote the paper: XMT JHC HYM TTL LZ.

## References

1. Zhang L, Wang Z, Sagotsky JA, Deisboeck TS (2009) Multiscale agent-based cancer modeling. *Journal of mathematical biology* 58: 545–559. doi: [10.1007/s00285-008-0211-1](https://doi.org/10.1007/s00285-008-0211-1) PMID: [18787828](https://pubmed.ncbi.nlm.nih.gov/18787828/)
2. Charles MM, Michael J. Tutorial on agent-based modeling and simulation part 2: how to model with agents; 2006. pp. 73–83.
3. Jacob C, Litorco J, Lee L (2004) Immunity through swarms: Agent-based simulations of the human immune system. *Artificial Immune Systems*: Springer. pp. 400–412.
4. Mansury Y, Diggory M, Deisboeck TS (2006) Evolutionary game theory in an agent-based brain tumor model: exploring the ‘genotype–phenotype’ link. *Journal of theoretical biology* 238: 146–156. PMID: [16081108](https://pubmed.ncbi.nlm.nih.gov/16081108/)
5. Siddiqi A, Niazi M, Mustafa F, Bokhari H, Hussain A, et al. A new hybrid agent-based modeling & simulation decision support system for breast cancer data analysis; 2009. IEEE. pp. 134–139.
6. Øksendal B (2003) *Stochastic differential equations*: Springer.
7. Jones DS, Plank M, Sleeman BD (2011) *Differential equations and mathematical biology*: CRC press.
8. Miao H, Xia X, Perelson AS, Wu H (2011) On identifiability of nonlinear ODE models and applications in viral dynamics. *SIAM review* 53: 3–39. PMID: [21785515](https://pubmed.ncbi.nlm.nih.gov/21785515/)
9. Miao H, Dykes C, Demeter LM, Wu H (2009) Differential equation modeling of HIV viral fitness experiments: model identification, model selection, and multimodel inference. *Biometrics* 65: 292–300. doi: [10.1111/j.1541-0420.2008.01059.x](https://doi.org/10.1111/j.1541-0420.2008.01059.x) PMID: [18510656](https://pubmed.ncbi.nlm.nih.gov/18510656/)
10. Folcik VA, An GC, Orosz CG (2007) The Basic Immune Simulator: An agent-based model to study the interactions between innate and adaptive immunity. *Theoretical Biology and Medical Modelling* 4.
11. Halling-Brown M, Pappalardo F, Rapin N, Zhang P, Alemani D, et al. (2010) ImmunoGrid: towards agent-based simulations of the human immune system at a natural scale. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 368: 2799–2815.
12. Fachada N, Lopes V, Rosa A. Agent-based modelling and simulation of the immune system: a review; 2007.

13. Chiacchio F, Pennisi M, Russo G, Motta S, Pappalardo F (2014) Agent-Based Modeling of the Immune System: NetLogo, a Promising Framework. *BioMed research international* 2014.
14. Miao H, Hollenbaugh JA, Zand MS, Holden-Wiltse J, Mosmann TR, et al. (2010) Quantifying the early immune response and adaptive immune response kinetics in mice infected with influenza A virus. *Journal of virology* 84: 6687–6698. doi: [10.1128/JVI.00266-10](https://doi.org/10.1128/JVI.00266-10) PMID: [20410284](https://pubmed.ncbi.nlm.nih.gov/20410284/)
15. Ho W-H, Chan AL-F (2011) Hybrid Taguchi-Differential Evolution Algorithm for Parameter Estimation of Differential Equation Models with Application to HIV Dynamics. *Mathematical Problems in Engineering* 2011: 14.
16. Kennedy J (2010) Particle swarm optimization. *Encyclopedia of Machine Learning*: Springer. pp. 760–766.
17. Kennedy J, Kennedy JF, Eberhart RC (2001) *Swarm intelligence*: Morgan Kaufmann.
18. Poli R (2007) An analysis of publications on particle swarm optimization applications. Essex, UK: Department of Computer Science, University of Essex.
19. Poli R (2008) Analysis of the publications on the applications of particle swarm optimisation. *Journal of Artificial Evolution and Applications* 2008: 3.
20. Clerc M (2012) Standard particle swarm optimisation.
21. Pedersen MEH, Chipperfield AJ (2010) Simplifying particle swarm optimization. *Applied Soft Computing* 10: 618–628.
22. Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) *Introduction to algorithms*: MIT press Cambridge.
23. Bang-Jensen J, Gutin G, Yeo A (2004) When the greedy algorithm fails. *Discrete Optimization* 1: 121–127.
24. Bendall G, Margot F (2006) Greedy-type resistance of combinatorial problems. *Discrete Optimization* 3: 288–298.
25. Ypma J (2013) *Introduction to SparseGrid*.
26. Heiss F, Winschel V (2008) Likelihood approximation by numerical integration on sparse grids. *Journal of Econometrics* 144: 62–80.
27. Yserentant H (2005) Sparse grid spaces for the numerical solution of the electronic Schrödinger equation. *Numerische Mathematik* 101: 381–389.
28. Nobile F, Tempone R, Webster CG (2008) A sparse grid stochastic collocation method for partial differential equations with random input data. *SIAM Journal on Numerical Analysis* 46: 2309–2345.
29. Cleveland WS, Grosse E, Shyu WM (1992) Local regression models. *Statistical models in S*: 309–376.
30. Cleveland WS (1981) LOWESS: A program for smoothing scatterplots by robust locally weighted regression. *American Statistician*: 54–54.
31. Cleveland WS (1979) Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association* 74: 829–836.
32. Cleveland WS, Devlin SJ (1988) Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American Statistical Association* 83: 596–610.
33. Nocedal J, Wright SJ (2006) *Least-Squares Problems*: Springer.
34. Weisstein EW (2002) *CRC concise encyclopedia of mathematics*: CRC press.
35. Xie X-f, Zhang W-j, Yang Z-l (2003) Overview of particle swarm optimization. *Control and Decision* 18: 129–134.
36. Miao H, Wu H, Xue H (2014) Generalized Ordinary Differential Equation Models. *Journal of the American Statistical Association* 109: 1672–1682. PMID: [25544787](https://pubmed.ncbi.nlm.nih.gov/25544787/)
37. Leiserson CE, Rivest RL, Stein C, Cormen TH (2001) *Introduction to algorithms*: MIT press.
38. Mitchell M (1998) *An introduction to genetic algorithms*: MIT press.
39. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions on* 6: 182–197.