

# Sample efficient likelihood-free inference for virus dynamics with different types of experiments

Yingying Xu<sup>\*†1,2,3</sup>, Ulpu Remes<sup>‡1,2</sup>, Enrico Rinaldi<sup>§3</sup>, and Henri Pesonen<sup>¶4</sup>

<sup>1</sup>Department of Mathematics and Statistics, University of Helsinki, Finland

<sup>2</sup>Finnish Center for Artificial Intelligence(FCAI), Finland

<sup>3</sup>RIKEN Center for Interdisciplinary Theoretical and Mathematical Sciences(iTHEMS), Japan

<sup>4</sup>Department of Biostatistics, University of Oslo, Norway

August 18, 2025

## Abstract

This study applied Bayesian optimization likelihood-free inference(BOLFI) to virus dynamics experimental data and efficiently inferred the model parameters with uncertainty measure. The computational benefit is remarkable compared to existing methodology on the same problem. No likelihood knowledge is needed in the inference. Improvement of the BOLFI algorithm with Gaussian process based classifier for treatment of extreme values are provided. Discrepancy design for combining different forms of data from completely different experiment processes are suggested and tested with synthetic data, then applied to real data. Reasonable parameter values are estimated for influenza A virus data.

**Keywords:** Bayesian optimization, likelihood-free inference, virus dynamics, influenza A, Gaussian process, simulator-based modeling

## 1 Introduction

In realistic mechanistical modelling, the full likelihood knowledge is often not available explicitly. In many cases, the state model itself is a simplification of an actual process, and the observation model is based on assumptions about the noise distribution. Given these models, a simulator is built with parameters to be inferred from observational data. As the number of parameters increases, the searchable parameter space becomes a large multidimensional space, and without the explicit likelihood function, the traditional inference techniques are unavailable. Likelihood-free inference techniques based on simulated data from the model do enable inference from an approximate model, but without enough prior knowledge of the parameters, the inference can require a large number of simulations with different parameter combinations. On many cases, this is infeasible, if the simulator is slow to execute. To address these difficulties, many intelligent algorithms are developed to sample the multidimensional parameter space efficiently and obtain the posterior distribution of the parameters quickly(Jarno Lintusaari *et al.*, 2017; Price *et al.*, 2017; Simola *et al.*, 2021; Ikonomov and Michael U. Gutmann, 2020). In this study, we employed Bayesian optimization likelihood-free inference(BOLFI) algorithm(Michael U Gutmann and Corander, 2016) using a Gaussian Process (GP) surrogate model for the discrepancy between simulated and observed data with active learning to iteratively propose where to sample the simulator based on the learned uncertainty. By learning the relationship of the parameters and the discrepancy, we can probabilistically estimate the parameter values for the problem. Based on the learned GP surrogate model, the likelihood function is obtained. Finally, the posterior distributions are obtained by sampling from the learned likelihood and input prior distributions.

---

<sup>\*</sup>yingying.xu@helsinki.fi

<sup>†</sup>ORCID: 0000-0002-9096-0552

<sup>‡</sup>ORCID: 0000-0003-1435-0207

<sup>§</sup>ORCID: 0000-0003-4134-809X

<sup>¶</sup>ORCID: 0000-0003-4500-2926

We applied BOLFI to influenza A virus dynamics experimental data to learn the six parameters in the ordinary differential equation model. The target parameters are listed in table I. Previous research employed comprehensive MCMC-based approaches, requiring extensive computation time (e.g. 43 hours on a cluster to analyze 500,000 parameter sets)(Quirouette *et al.*, 2024). Moreover, this learning is based on small scale datasets. To improve the efficiency of learning, we applied the BOLFI algorithm to the same data sets and the same simulated data generating models with arrangement of combination. Remarkably, BOLFI completed this task with only 1 hour and 26 minutes for surrogate model learning, plus the sampling process, in total it is about 2 hours to obtain the posterior distributions for all six parameters, representing a 21.5-fold efficiency gain. Notably, our method accomplishes this without relying on explicit likelihood functions, thus frees researchers from extensive efforts on likelihood examinations previously necessary.

The contributions of this study are as follows.

- We applied Bayesian optimization likelihood-free inference(BOLFI) to virus dynamics experimental data and efficiently inferred the six model parameters with uncertainty measure.
- Total learning time by BOLFI is total it is about 2 hours which is 21.5 times faster than the existing study with MCMC.
- No likelihood knowledge is needed in the inference.
- Improvement of the BOLFI algorithm with Gaussian process based classifier for treatment of extreme values are provided.
- Discrepancy design for combining different forms of data from completely different experiment processes are suggested and tested with synthetic data, then applied to real data.
- Reasonable parameter values are estimated for influenza A virus data.

The rest of the paper is organized as: section 2 is problem setting where we explain the simulator, how the two different types of experimental data are obtained and what target parameters are to be learned; section 3 explains the inference method BOLFI algorithm, why the classifier is needed and how we integrated it with BOLFI, how we designed discrepancy for this specific problem combining two types of forms of data from four experiments; section 4 shows the learning result for parameters including test with synthetic data and experimental data; section 5 discusses about the meaning of the results and what we can learn from the study.

## 2 Problem setting

Parameter	Symbol
rate of one infectious virion enters a cell and successfully infects it	$\gamma$
rate of infectious virion irreversible entry a cell	$\beta$
infectious production rate for total RNA virions	$P_{RNA}$
infectious production rate for infectious virions	$P$
mean duration for the eclipse phase	$\tau_E$
mean duration for the infectious phase	$\tau_I$

Table I: Parameters to be inferred

We are looking at the virus cell infection process which involves several key steps that allow the virus to enter the host cell, replicate, and produce new viral particles. Figure1 shows the virus cell infection process with rate parameters (table I,II) corresponding to each steps.

The model is governed by a system of ordinary differential equations (ODEs), with six unknown parameters: the virion entry rate  $\beta$ , infection probability  $\gamma$ , production rate of total viral RNA  $P_{RNA}$ , production rate of infectious virions  $P$ , and the durations of the eclipse phase  $\tau_E$  and infectious phase  $\tau_I$ . Each of these parameters corresponds to a distinct mechanistic process in the viral replication cycle, and their biological interpretation is visually anchored in Figure 1.

In the modeling framework, the observable variable VRNA (viral RNA) includes all RNA species measured from the experiment, which encompasses both infectious virions (denoted as  $V$ ) and non-infectious viral RNA strands. Therefore,  $V$  is considered a subset of VRNA. The production of VRNA is governed by parameters such as  $P_{RNA}$  (total RNA production rate) and  $c_{RNA}$  (degradation factor), which indirectly influence  $V$  through shared replication mechanisms.

Symbol	Parameter
$T$	target cells
$V$	infectious virus
$V_{RNA}$	total viral RNA
$E_i, i = 1, 2, \dots, n_E$	the number of cells in the $i$ th eclipse compartment
$I_i, i = 1, 2, \dots, n_I$	the number of cells in the $i$ th infectious compartment
$n_E$	total number of eclipse compartments
$n_I$	total number of infectious compartments
$\tau_E$	length of eclipse phase
$\tau_I$	length of infectious phase
$c$	infectious virus clearance rate
$c_{RNA}$	total virus clearance rate

Table II: Additional model parameters

This distinction is important because the RNA measurements from experiments quantify total viral RNA regardless of infectivity, while the infectious virion data ( $V$ ) correspond only to a functionally active subset. See figure 2 for visual explanation.

By combining experimental data of two types—quantitative RNA measurements and infectious virion counts—from four different experimental conditions, we aim to infer these six parameters using a simulation-based inference framework.

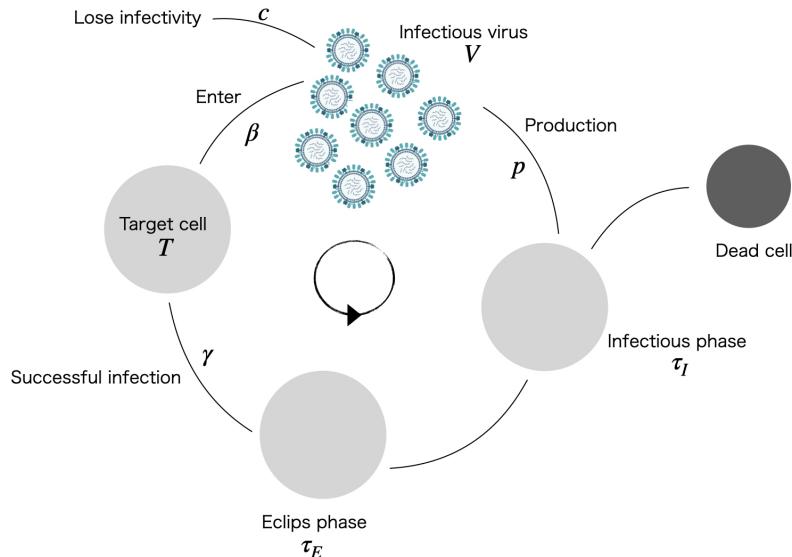


Figure 1: virus cell infection process

## 2.1 Experimental data

We analyze the data from (Simon *et al.*, 2016). For both Single Cycle and Multiple Cycle infection experiments, we have time steps samples of both total viral RNA concentration and Endpoint Dilution(ED) assay experiments.

The viral RNA concentration data are the measurements that quantify the amount of viral RNA present in a sample. This data typically indicates the level of viral genetic material, which can be used to assess the severity of infection, viral replication activity, and the effectiveness of treatments or interventions. This data is often obtained through techniques like quantitative PCR (qPCR) or digital PCR, which amplify and quantify specific viral RNA sequences. Viral RNA concentration data is a real number with unit vRNA/ml.

ED assay experiment is used to measure the amount of infectious virus in a sample (Reed and Muench, 1938; D. Cresta *et al.*, 2021). In the experiment, the sample was serially diluted and cell cultures were prepared with each dilution. After incubation, assess whether the cells show signs of infection. They determine the highest dilution of a sample at which the virus can still infect cells and produce a detectable effect. This method allows researchers to calculate the 50% infectious

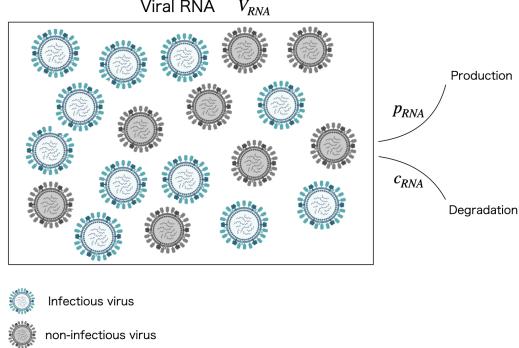


Figure 2: viral RNA( $V_{RNA}$ ) includes infectious virus and non-infectious virus RNA. There is a hierarchical relationship between  $V_{RNA}$  and  $V$ .

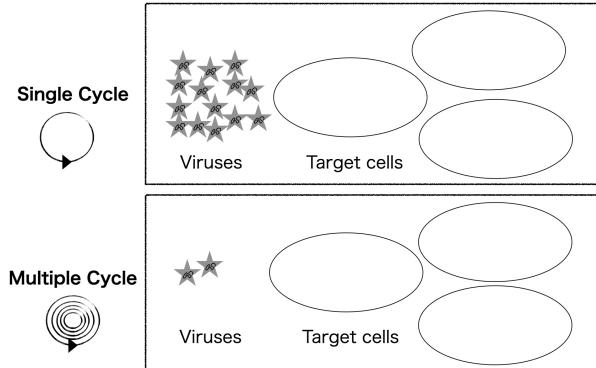


Figure 3: Image of single cycle and multiple cycle infection experiment. Single cycle experiment started with many virus, therefore, all target cells can be infected at once in one cycle. In contrast, in multiple cycle experiment, in the beginning only very few viruses are prepared, therefore, it takes multiple cycles of infection, reproduce new viruses and get infected again in several rounds. The difference between two experiments is the starting virus volume. In the text, they are abbreviated as SC and MC respectively for convenience.

dose (TCID<sub>50</sub>), which is a standard measure of viral infectivity. In our study, we are not using TCID<sub>50</sub>; instead, we use the infectious numbers from the ED assay outcome count directly as data for parameter inference. In our case, Each ED assay experiment output is a 8 digit vector with integers(0, 1, 2, 3, 4) as elements. Because in this case, the ED assay experiments are done with 8 columns with increasing delusion factors and each column has 4 replica of the same conditional tube. The outcome vector of the ED assay experiment is the infected tube count in each column of the ED assay experiment.

We have two settings of experiments, the single-cycle(SC) infections and the multiple-cycle(MC) infections as described in (Simon *et al.*, 2016). Single cycle experiment started with many virus, therefore, all target cells can be infected at once in one cycle. In contrast, in multiple cycle experiment, in the beginning only very few viruses are prepared, therefore, it takes multiple cycles of infection, reproduce new viruses and get infected again in several rounds. See figure 3 for the image which shows the difference of the starting volume of the virus are the main difference between SC and MC experiments. In the SC experiment, the virus was inoculated onto the cells at  $t = -1$  h to achieve a MOI of 3 within one hour, and the cell culture was rinsed at time  $t = 0$  h to remove any virus still present from the high inoculum. The MC infection is initiated at time  $t = 0$  h with no rinse thereafter. Over the course of both the SC and MC experimental infections, at each sampling time, 0.5 ml of the 10 ml supernatant is removed for virus quantification and replaced with 0.5 ml of fresh media.

Therefore, we have four data sets in total. Figure 4 shows the corresponding measurement. The four data sets, SC RNA data, MC RNA data, SC ED data, MC ED data are listed in table 13, 15, table VI, V and visualized in figures 5, 6 and 7. See caption of the figures for detailed observations

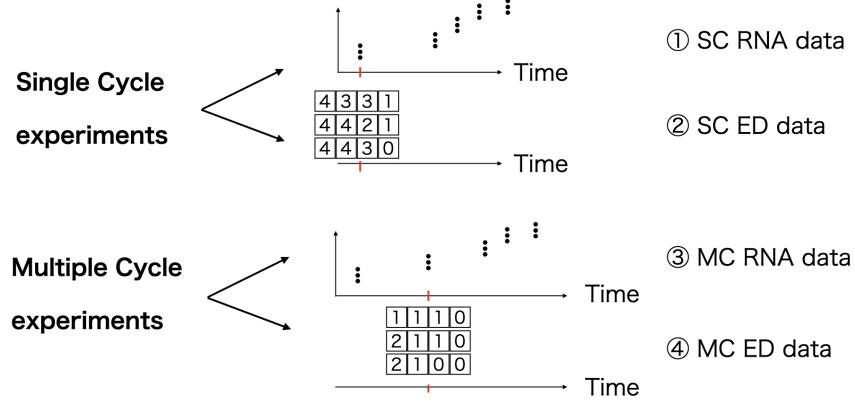


Figure 4: This figure is a data measurement image. Since it is a conceptual image, the data points number, integers and time steps are only a example, do not show the real data we used. For real data set, please refer to table 13, 15, table VI, V. There are four data sets in total: singe cycle RNA data, single cycle ED data, multiple cycle RNA data and multiple cycle ED data. For both Single Cycle and Multiple Cycle infection experiments, there are 12 time steps measurements. For each time step measurement, we have both total viral RNA concentration test result and ED assay experiment outcomes. Viral RNA concentration data is a real number with unit vRNA/ml. Each ED assay experiment output is a vector with integers as elements. The outcome vector of ED assay experiment is the infected tube count in each column of the ED assay experiment. There are several replicates of single cycle and multiple cycle experiments which give the several measurements fo the same time steps in the RNA and ED data sets. Generally there are 3 replicates in ED data (figure 6 and 7). In RNA data, there are time steps with irregular replicate numbers as 2 or 6 which can be observed in data table 14, 16 and figure 5.

of the data sets. There are several repeats of single cycle and multiple cycle experiments which give the several measurements for the same time steps in the RNA and ED data sets. Generally, there are 3 repeats in ED data. In RNA data, there are time steps with irregular repeat numbers as 2 or 6.

The time steps are the same for ED and RNA data outcome from The MC experiment. The same is true for the SC experiment, which has ED and RNA data output with the same time steps. However, the MC and SC experiments have different measurement times, but both have 12 time steps. The measuring times are listed in table 13, 15, table VI, V.

## 2.2 The simulator

Employing the two simulators in (Quirouette *et al.*, 2023), one describing the virus dynamics over time by a ordinary differential equation(ODE) and one simulating the ED assay experiment, we created a simulator which outputs four synthetic data sets(SC RNA data, SC ED data, MC RNA data and MC ED data) that have the same format as the experimental data that we are going to analyze. The ED assay simulator is stochastic, however, the ODE itself is deterministic. To make it a stochastic simulator, we added Gaussian random noise to the ODE simulated output and the variances of the noise are estimated from the experimental data to be learned. Adding noise to RNA measurements by scaling the RNA volume and applying a log-normal noise based on the specified standard deviation:  $\sigma_{MCRNA} = 0.226$  characterizes variability in multiple-cycle total virus data;  $\sigma_{SCRNA} = 0.145$  characterizes variability in single-cycle total virus data. In (Quirouette *et al.*, 2023), the ODE simulator and the ED assay simulator both only outputs one set of time series data. In contrast, we created a simulator repeats the simulation of one parameter set for 6 times(because the largest repeats number in both the RNA and ED experimental data is 6), then randomly select the same replica number for each data sets in each time step according to the experimental data from the 6 sets of time series to create a simulated data has the same format of experimental data. Table IV in (Quirouette *et al.*, 2023) summarizes all the fixed parameters and initial conditions used in simulating these infections. In appendix, A and B, we summarized

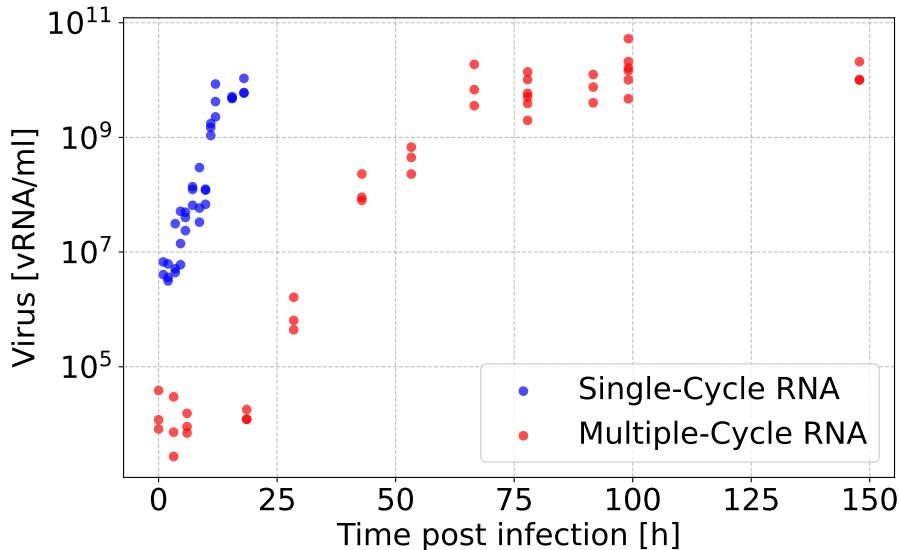


Figure 5: Putting Single Cycle and Multiple Cycle virus RNA data together. One can observe that in SC experiment cells reach the all infected saturation state in a short time compare to MC experiment, because the starting amount of viruses are much more than in MC experiments. Refer to table 13, figure 14, table 15 and figure 16 for detailed observation. Data is from (Simon *et al.*, 2016).

the main explanation of the ODE simulator and ED assay simulator in (Quirouette *et al.*, 2023). Given the experimental data sets and the simulator for the four data sets, our goal is to infer the value of the parameters with uncertainty,  $\gamma, \beta, p, p_{RNA}, n_I, n_E$  in the model.

### 2.2.1 Units

In the simulation in this study, we use [IV] unit for  $V_o$ . There are two units in previous study (Quirouette *et al.*, 2024)– [SIN] unit and [IV] unit. The initial SIN unit is converted to IV unit using,

$$C_{\text{measured}} = C_{\text{actual}} \cdot P_{V \rightarrow \text{Establishment}} \quad (1)$$

in both SC and MC infections. This equation relates the experimentally measured concentration of infection-causing units (SIN/ml) to the actual infectious virion concentration (IV/ml) by considering the probability of a successful infection establishment(see eq (17)) (Quirouette *et al.*, 2024).

## 3 The likelihood free inference method

### 3.1 The BOLFI algorithm(Michael U Gutmann and Corander, 2016)

When only limited information is available about relevant regions of the parameter values, a large number of simulations via the forward model are typically required. Although the simulator used in this study is relatively efficient, the overall computational cost of the MCMC algorithm remains significant (Quirouette *et al.*, 2024).

To mitigate the need for numerous potentially expensive simulations, active learning methods adapt the querying process based on different strategies. BOLFI employs Bayesian optimization (Michael U Gutmann and Corander, 2016) to iteratively build a probabilistic surrogate model of the relationship between parameters  $\theta$  and discrepancies  $d(\theta) = d(s(y), s(y_{\text{obs}}))$ , where  $s(\cdot)$  is the summary statistics function over simulated data  $y$  and observation data  $y_{\text{obs}}$  and  $d(\cdot)$  represents a distance measure function, using the growing evidence set  $\mathcal{E}_t$ , which contains pairs  $\{(\theta_i, d(\theta_i))\}$  for  $(i = 1, \dots, t)$ .

The original formulation of BOLFI uses a Gaussian process (GP) as the surrogate model for the discrepancy function, and new evidence  $\{(\theta_{t+1}, d(\theta_{t+1}))\}$  is sampled from relevant regions of the parameter space. Relevant regions are determined to be parts of the space where the discrepancy

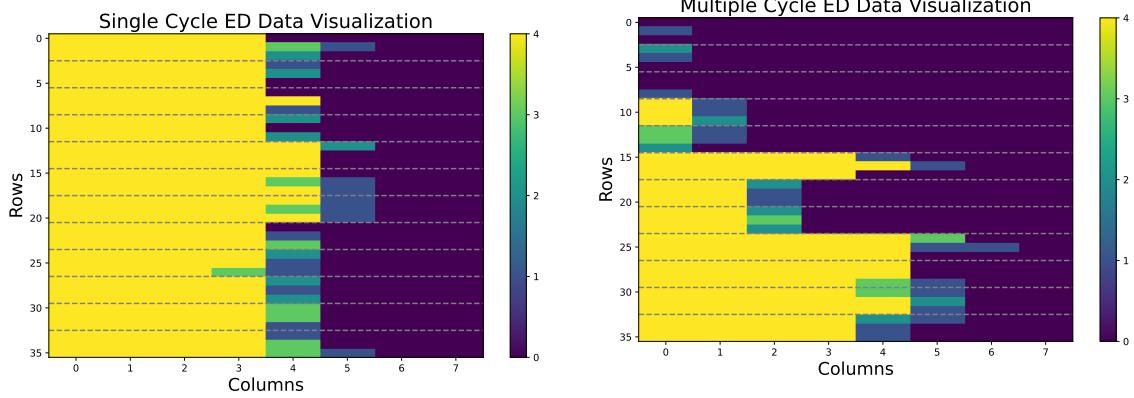


Figure 6: Visualization of single cycle ED data as a color map. The Grey lines in between are separating each time step's data. One can observe that in each time step, the ED assay output variate in the last few columns when the infected number is getting small and close to zero which means no tube is infected in that column. Since in single cycle experiment, it started with many viruses, the first measurement in early time is already with many columns are fully infected. The change in time is not very big comparing to multiple cycle experiment. In this experiment, the replicate number in each column is 4. Note, referring to table 13, the delusion factor are the same for time  $1 \sim 8.62$  hours post-infection. One order of shift in delusion factor in settings for time  $9.9 \sim 18$  hours post-infection(the last five steps). Data is from (Simon *et al.*, 2016).

Figure 7: Visualization of multiple cycle ED data as a color map. The Grey lines in between are separating each time step's data. Note, referring to table 15, the delusion factors are increased for two to three orders in time steps  $28.5, 53.3, 66.5833$  hours post-infection which are the 5th and 7th, 8th measurement blocks in the figures counted from the top. One can observe that in multiple cycle experiments, as time passes, the infected number of tubes are increasing in general. More changes are seen comparing to SC experiments. This observation consists with the expected multiple cycle of infection from viruses to cells and reproduced viruses, then more cells are getting infected in the cycles. The color representation is the same with the left figure. The replicate number in each column is 4. Therefore, 4 in the data means all tubes are infected in that column. Data is from (Simon *et al.*, 2016).

is small. The probabilistic model/posterior is defined as

$$d(\theta) | \mathcal{E}_t \sim \text{GP} (\mu_t(\theta), v_t(\theta) + \sigma^2), \quad (2)$$

where GP is a Gaussian process with mean and variance functions,

$$\mu_t(\theta) = \mathbf{k}_t(\theta)^T \mathbf{K}_t^{-1} \mathbf{d}_t, \quad (3)$$

$$v_t(\theta) + \sigma^2 = k(\theta, \theta) - \mathbf{k}_t(\theta)^T \mathbf{K}_t^{-1} \mathbf{k}_t(\theta) + \sigma^2. \quad (4)$$

where  $\mathbf{k}_t(\theta)$  is a vector  $[k(\theta, \theta_1) \dots k(\theta, \theta_t)]^T$ , and  $\mathbf{K}_t$  is a matrix with elements  $k(\theta_i, \theta_j)$ , both defined via covariance functions  $k(\theta', \theta'')$ . A common choice for  $k(\theta', \theta'')$  is the squared exponential covariance function:

$$k(\theta', \theta'') = \sigma_f^2 \exp \left( -\sum_{j=1}^d \frac{(\theta'_j - \theta''_j)^2}{\lambda_j^2} \right), \quad (5)$$

where  $\lambda_j$  are the length scale parameters. When fitting a GP to the evidence set, the hyperparameters  $\sigma$ ,  $\sigma_f$ , and  $\lambda_j$  can be iteratively optimized (Rasmussen and Williams, 2006).

From  $d(\theta)$ , a suitable pointwise approximation to the likelihood function is retrieved as

$$L(\theta) \approx \Phi \left( \frac{h - \mu_t(\theta)}{\sqrt{v_t(\theta) + \sigma^2}} \right), \quad (6)$$

where  $\Phi(\cdot)$  is the Gaussian cumulative density, and  $h$  is a threshold parameter chosen as the minimum of  $\mu_t(\theta)$ .

In practice, an acquisition function determines the locations of the relevant parameter space, and there are several reasonable choices. One example is the lower confidence bound selection criterion (LCBSC):

$$A_t(\theta) = \mu_t(\theta) - \sqrt{\eta_t^2 v_t(\theta)}, \quad (7)$$

where  $\eta_t^2$  depends on the iteration  $t$ , the parameter space dimension  $d$ , and other tunable parameters (Srinivas *et al.*, 2010). The next parameter value to sample is obtained by minimizing (7) and randomly varying it to further balance the exploration and exploitation of the probabilistic function of the discrepancy.

### 3.2 BOLFI with Classifier(failure robust)–Treatment of extreme values

The simulator in (Quirouette *et al.*, 2023) returns none output for invalid (physically not possible or not satisfying certain constraints) parameter combinations. In practice, the simulator uses several decision rules to check intermediate simulation results and decide whether to continue or abort the simulation. The rules can be viewed as hidden parameter constraints. While known parameter constraints such as  $P_{RNA} > P$  by definition can be encoded in the prior distribution in BOLFI to avoid invalid parameter combinations and aborted simulations, this is not possible with the hidden constraints that are implemented in the simulator from (Quirouette *et al.*, 2023) and not known in advance.

The hidden constraints and aborted simulations are a problem in BOLFI because without the simulated output  $y'$ , it is not possible to calculate  $d(\theta')$  and update the surrogate model as usual. Meanwhile, it is also not advisable to skip the model update and not learn from the simulation result at all. Since BOLFI uses the surrogate model and acquisition rules to choose the parameter values considered in the next simulation, no model update could mean that BOLFI repeatedly chooses invalid parameter combinations in an attempt to learn about discrepancies that cannot be observed. To prevent this, BOLFI should learn to avoid invalid parameter combinations that result in aborted simulations.

The experiments carried out in this work use BOLFI extended with a classifier. The approach is similar to the one introduced by El Gammal *et al.*, (2023). The idea is to combine the regression model that learns dependencies between simulator parameters and discrepancies with a binary classifier that learns the hidden constraints that divide parameters into valid and invalid combinations. While the regression model is only updated when the simulation succeeds and returns an output  $y$ , the classifier is updated based on all attempted and succeeded simulations, and predicted discrepancies are calculated based on the combined model. In practice, if the parameters  $\theta$  are classified as valid, the predicted mean and variance are calculated based on Equations (3) and (4) as usual, and if the parameters are classified as invalid, the predicted mean  $\mu(\theta) = \infty$  and predicted variance  $\sigma(\theta) = 0$ .

El Gammal *et al.*, (2023) combined GP regression with a support vector machine classifier, whereas previous works on BO under hidden constraints have commonly used GP classifiers. For a review, see (Bussemaker *et al.*, 2024). The BOLFI version studied in this work uses GP models for both the classifier and the regression component.

In Figure 20, the vertical magenta lines indicate failed simulation cases. These failures occur with decreasing frequency as the learning process advances. This trend demonstrates that the classifier successfully guides the algorithm away from regions of the parameter space associated with simulation failures. As a result, the occurrence of failed simulations diminishes progressively throughout the inference process.

The steps of the BOLFI with classifier algorithm are summarized in Algorithm 1, and its implementation in ELFI software (Lintusaari *et al.*, 2018) is used in this study. Here is a demo notebook with a 2D toy example to show how it works:(Remes, 2024). The algorithm is robust in the sense that it automatically handles outputs that are invalid or extreme.

---

**Algorithm 1** BOLFI with classifier

---

```

1: Given simulator  $p(y|\theta_t)$  with parameters as elements in vector  $\theta$ 
2: Define discrepancy function  $d$  and prior distribution  $p(\theta)$  for parameters
3: Initialization:
4: for  $t = 1, \dots, N_{\text{init}}$  do
5:   Generate  $\theta_t \sim p(\theta)$ 
6:   if the simulator is valid for  $\theta_t$  then
7:     GP classifier is classified as valid
8:     Simulate  $y' \sim p(y|\theta_t)$ 
9:     Calculate  $d_t = d(s(y_{\text{obs}}), s(y'))$ 
10:    end if
11:   if the simulator is invalid for  $\theta_t$  then
12:     GP classifier is classified as invalid
13:   end if
14: end for
15: Set  $\mathcal{E}_{N_{\text{init}}} = \{(\theta_t, d_t)\}_{t=1}^{N_{\text{init}}}$ 
16: Fit  $d(\theta) | \mathcal{E}_{N_{\text{init}}} \sim \text{GP}(\mu_{N_{\text{init}}}(\theta), v_{N_{\text{init}}} + \sigma^2)$ 
17: in the invalid region, set the predicted GP mean  $\mu(\theta) = \infty$  and predicted variance  $\sigma(\theta) = 0$ 
18:
19: Learning:
20: for  $t = N_{\text{init}} + 1, \dots, N_E$  do
21:   if  $t \equiv T_{\text{update}} \bmod T_{\text{update}}$  then
22:     Optimize GP hyperparameters
23:   end if
24:   Calculate  $\theta_t = \arg \min A_t(\theta)$ 
25:   Generate  $\theta_t \sim \mathcal{N}(\theta_t, \sigma_{\text{acq}}^2)$ 
26:   if the simulator is valid for  $\theta_t$  then
27:     GP classifier is classified as valid
28:     Simulate  $y' \sim p(y|\theta_t)$ 
29:     Calculate  $d_t = d(s(y_{\text{obs}}), s(y'))$ 
30:     Update  $\mathcal{E}_t = \mathcal{E}_{t-1} \cup \{(\theta_t, d_t)\}$ 
31:     Fit  $d(\theta) | \mathcal{E}_t \sim \text{GP}(\mu_t(\theta), v_t + \sigma^2)$ 
32:   end if
33:   if the simulator is invalid for  $\theta_t$  then
34:     GP classifier is classified as invalid
35:     in the invalid region, set the predicted GP mean  $\mu(\theta) = \infty$  and predicted variance
 $\sigma(\theta) = 0$ 
36:   end if
37: end for
38:
39: Output:
40: Calculate likelihood function  $L(\theta)$  as in Equation (6)
41: Calculate corresponding posterior distribution with prior  $\pi(\theta|y_{\text{obs}}) = p(\theta)L(\theta)$ 
42: Draw samples from the learned posterior distribution  $\pi(\theta|y_{\text{obs}})$ 

```

---

### 3.3 Setting

Adopting the same scale settings from previous studies, we set the parameter  $\gamma$ ,  $P$ ,  $\text{beta}(\beta)$ ,  $P_{RNA}$  to log scale as  $\hat{\gamma} = \log_{10} \gamma = 10^\gamma$ ,  $\hat{P} = \log_{10} P = 10^P$ ,  $\hat{\beta} = \log_{10} \beta = 10^\beta$ ,  $\hat{P}_{RNA} = \log_{10} P_{RNA} = 10^{P_{RNA}}$ . We keep  $\tau_E$  and  $\tau_I$  on a linear scale.

Parameter	Distribution	Range	Constraints
$P$	Uniform (log scale)	0.5 to 3	Subject to $P < P_{RNA}$
$P_{RNA}$	Uniform (log scale)	2 to 4.8	Subject to $P < P_{RNA}$
$\gamma$	Uniform (log scale)	-1 to 0.5	—
$\beta$	Uniform (log scale)	-8 to -6	—
$\tau_E$	Uniform (linear scale)	2 to 15	—
$\tau_I$	Uniform (linear scale)	2 to 55	—

Table III: Prior distributions for model parameters.

Prior settings for parameters in the model are listed in table III. We added a condition for  $P$  and  $P_{RNA}$  as  $P < P_{RNA}$ . This condition is based on the definition of the parameters as  $V$  is within  $V_{RNA}$ , see figure 2. First, we set prior for  $P$  to be in log scale in bounds (0.5, 3). Prior for  $P_{RNA}$  is in log scale in bounds (2, 4.8). Then we update parameter bounds adaptively during inference to create a dynamic prior distributions. We take the minimum value of  $\hat{P}_{RNA}$  as the upper bound of  $\hat{P}$  adaptively. This helps focus inference within a relevant parameter space, avoiding invalid regions to be more efficient. Figure 8 shows this constraint as “scale\_p” which is the adaptive range of  $\hat{P}$  and “upper\_p” is the minimum value of  $\hat{P}_{RNA}$ .

### 3.4 Discrepancy

In likelihood-free inference, the discrepancy function is a measure used to quantify the difference between the simulated data generated under a set of parameters and the observed data. Since the true likelihood is intractable or computationally expensive to evaluate directly, the discrepancy serves as a proxy to assess how well a parameter set reproduces the observed data. Typically, the discrepancy  $d(\theta)$  is defined based on summary statistics of the data, such as:

$$d(\theta) = |s(y_{\text{sim}}(\theta)) - s(y_{\text{obs}})| \quad (8)$$

where:  $y_{\text{sim}}(\theta)$  is the simulated data generated using parameters  $(\theta)$ ,  $y_{\text{obs}}$  is the observed data,  $s(\cdot)$  denotes summary statistics extracted from the data (e.g., mean, variance, autocorrelation),  $|\cdot|$  is a norm (often Euclidean or  $L_1$ ) measuring the difference between summaries. In practice, the discrepancy provides a way to approximate how close the simulated data are to the observed data, guiding the inference process toward parameter values that minimize this discrepancy, thus indicating a better fit.

In this work, the discrepancy is illustrated in a digraph in figure 8. In the figure, “list\_output” represents the output of the simulator. The nodes  $s_1, s_2, s_3$  and  $s_4$  represents MC ED data, MC RNA data, SC ED data, and SC RNA data, respectively. “failed” means the situation when the simulator fails and outputs “none”.  $d_1$  calculates  $L_1$  distance between simulated and observed MC ED data and normalize it with 36( $3 \times 12$ ),  $d_1 = \frac{1}{36} \sum_{i=1}^{36} |y_{\text{sim,MC\_ED},i} - y_{\text{obs,MC\_ED},i}|$ .  $d_3$  is the  $L_1$  distance between simulated and observed SC ED data and normalize it with 36,  $d_3 = \frac{1}{36} \sum_{i=1}^{36} |y_{\text{sim,SC\_ED},i} - y_{\text{obs,SC\_ED},i}|$ . Normalization makes this  $L_1$  distance comparable with the Euclidean distance for RNA data. The reason for this normalization constant 36 is because the ED data contains 12 time steps and each step has three duplicates, therefore, in total there are 36 data points in ED data for both MC and SC experiments.  $d_{13}$  performs element-wise addition between the outputs of nodes  $d_1$  and  $d_3$ ,  $d_{13} = d_1 + d_3$ .  $c_2$  and  $c_4$  computes the base-10 logarithm of the MC RNA data and the SC RNA data, respectively. They are used as summary statistics. Summary statistics are used to reduce the dimensionality of the data or to extract meaningful features, which can simplify likelihood-free inference.  $d_2$  and  $d_4$  calculates the Euclidean distance between the simulated and observed logarithm transformed MC RNA data( $c_2$ ) and the SC RNA data( $c_4$ ), respectively. Therefore,  $d_2 = \|(\log_{10}(y_{\text{sim,MC RNA}})) - \log_{10}(y_{\text{obs,MC RNA}})\|_2$ ,  $d_4 = \|(\log_{10}(y_{\text{sim,SC RNA}})) - \log_{10}(y_{\text{obs,SC RNA}})\|_2$ .  $d_{24}$  add the outputs of nodes  $d_2$  and  $d_4$  together.  $d$  add the outputs of nodes  $d_{24}$  and  $d_{13}$  together.  $d_{\text{norm}}$  is normalized distance using the formula  $(d - \text{mean})/\text{std}$ . Here, the mean is estimated as 20 and the standard deviation was estimated as 10 using a simulated distribution of distance  $d$ . The purpose of normalization(Scaling) discrepancy is to make them easier to compare or combine with other nodes, especially if they are on different scales. This is a kind of standardization that ensures consistency in distance measures, helping inference algorithms converge more effectively.  $d_{\text{masked}}$  represents a masking operation for normalized discrepancy values, ensuring that certain elements corresponding to failures are replaced with a predefined value, such as infinity.

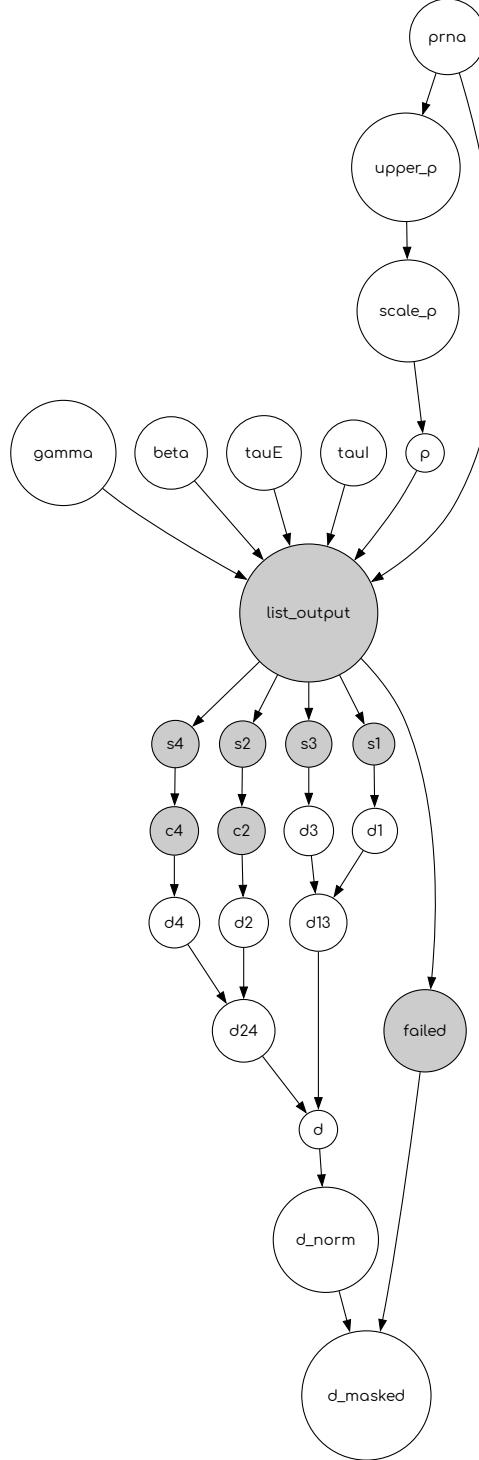


Figure 8: The digraph of the inference setting by BOLFI+classifier describing input parameters, output of simulators and discrepancy definition.  $d_{\text{masked}}$  is used as the final discrepancy in inference. Note, gamma, beta, p, prna in the figure is representing  $\log_{10} \gamma, \log_{10} \beta, \log_{10} P, \log_{10} P_{RNA}$ , respectively. tauE is  $\tau_E$  and tauI is  $\tau_I$  which are on linear scale.

## 4 Results

### 4.1 Test result by synthetic data

We tested the proposed inference procedure using synthetic data generated with the simulator and fixed set of parameters as the observed data. The result is shown in Figure 9. The lowest values of

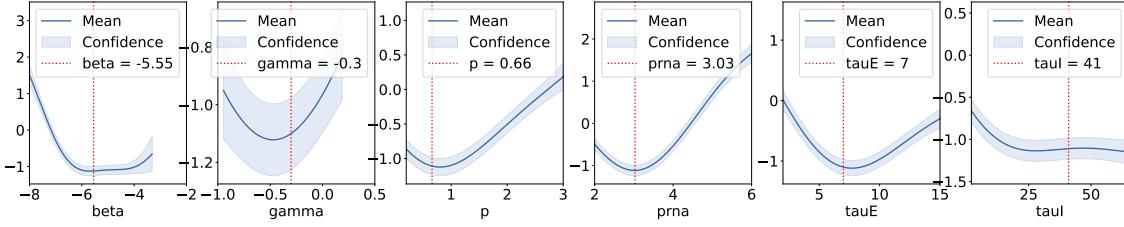


Figure 9: A series of one-dimensional slices of the learned multidimensional Gaussian Process (GP) surrogate model (the target model in BOLFI) are shown for each parameter dimension, generated using synthetic data produced with default parameter values (the red dashed vertical lines). The solid lines represent the mean estimates, while the shaded regions indicate the confidence intervals around these estimates. The fixed point used for plotting are posterior sample mean by 50000 mcmc samples. The values are [-5.406(ml/(cell·h)), -0.379(cell/[V]), 0.913([V]/(cell·h)), 3.056(vRNA/(cell·h)), 7.436(h), 38.522(h)] for each parameter in the order of the input of the simulator. Note, gamma, beta, p, prna in the figure is representing  $\log_{10} \gamma$ [ml/(cell·h)],  $\log_{10} \beta$ [cell/V],  $\log_{10} P$ [V/(cell·h)],  $\log_{10} P_{RNA}$ [vRNA/(cell·h)], respectively. The parameters  $\tau_E(\tau_E)$  and  $\tau_I(\tau_I)$  are on a linear scale with unit [h].

the surrogate model along each one-dimensional slice of the parameters are close to or correspond to the true parameter values used to generate the synthetic data. This suggests that our inference method is effective and capable of accurately recovering the true parameters.

#### 4.2 Learning result from experimental data

The inference was performed using 1,000 simulations, resulting in 759 valid evidence points and 241 failed outputs. The BOLFI algorithm took approximately 1 hour and 26 minutes to learn the target model on the CSC PUHTI computer. Sampling from the posterior distribution using a Metropolis sampler required about 25 minutes for 100,000 samples. Using the discrepancy measure defined in Section 3.4 and Figure 8, the results are presented in Figures 10, Figure 17, 19, 18, 20, and 12. Figure 10 depicts the distribution of discrepancies for each parameter during inference. It can be observed that the samples are concentrated around the regions with minimal discrepancy values, which correspond to the true or optimal parameter estimates. Figure 10 shows a matrix of pairwise plots from the Gaussian Process (GP) regression models, illustrating the relationships between parameters and their associated discrepancies. The regions excluded by the classifier indicate parameter spaces where the simulator fails or is highly likely to fail. Figure 17 presents one-dimensional slices of the learned multidimensional GP surrogate model for each parameter. The minima of these slices correspond to the optimal parameter estimates. Figure 19 displays two-dimensional contour plots of the predicted surrogate function across all pairs of parameters, providing a visual of the model's behavior in the multidimensional parameter space.

Table IV: Parameters estimated by BOLFI algorithm<sup>a</sup>

Parameter(unit)	Mean [95% CI]
$\tau_E$ (h)	8.142 [4.781, 11.472]
$\tau_I$ (h)	29.348 [5.616, 53.241]
$p_{RNA}$ (vRNA/(cell·h))	$10^{3.231[2.604, 3.961]} \sim 1702.1585[401.7908, 9141.1324]$
$p$ ([V]/(cell·h))	$10^{1.690[0.850, 2.611]} \sim 48.9779[7.0795, 408.3194]$
$\gamma$ (cell/[V])	$10^{-0.346[-0.703, -0.006]} \sim 0.4508[0.1982, 0.9863]$
$\beta$ (ml/(cell·h))	$10^{-6.818[-7.734, -6.094]} \sim 1.5205 \times 10^{-7}[1.8450 \times 10^{-8}, 8.0538 \times 10^{-7}]$

<sup>a</sup> Estimates are provided as: Mean [95% CI] where the mode and 95% Credible Interval(CI) correspond to the Marginal Posterior Distribution(MPD) for each parameter, marginalized over all others. 95% CI is represented as [0.025 0.975] quantiles. The Medians(0.5 quantile) of each parameter are  $\tau_E = 8.16$ (h),  $\tau_I = 28.39$ (h),  $p_{RNA}=10^{3.22}$ (vRNA/(cell·h)),  $p=10^{1.67}$ ([V]/(cell·h)),  $\gamma=10^{-0.35}$ (cell/[V]),  $\beta=10^{-6.78}$ (ml/(cell·h)). For the Maximum A Posteriori(MAP) which corresponds to the parameter set with the highest likelihood in the joint, multi-dimensional posterior  $P_{post}(\text{data})$ , the learned result is  $\tau_E = 7.898$ (h),  $\tau_I = 19.109$ (h),  $p_{RNA}=10^{3.309}$ (vRNA/(cell·h)),

$p=10^{1.414}([V]/(\text{cell}\cdot\text{h}))$ ,  $\gamma=10^{0.007}(\text{cell}/[V])$ ,  $\beta=10^{-6.958}(\text{ml}/(\text{cell}\cdot\text{h}))$ . We used MFM simulator with added Gaussian noise and combined with ED data simulator.

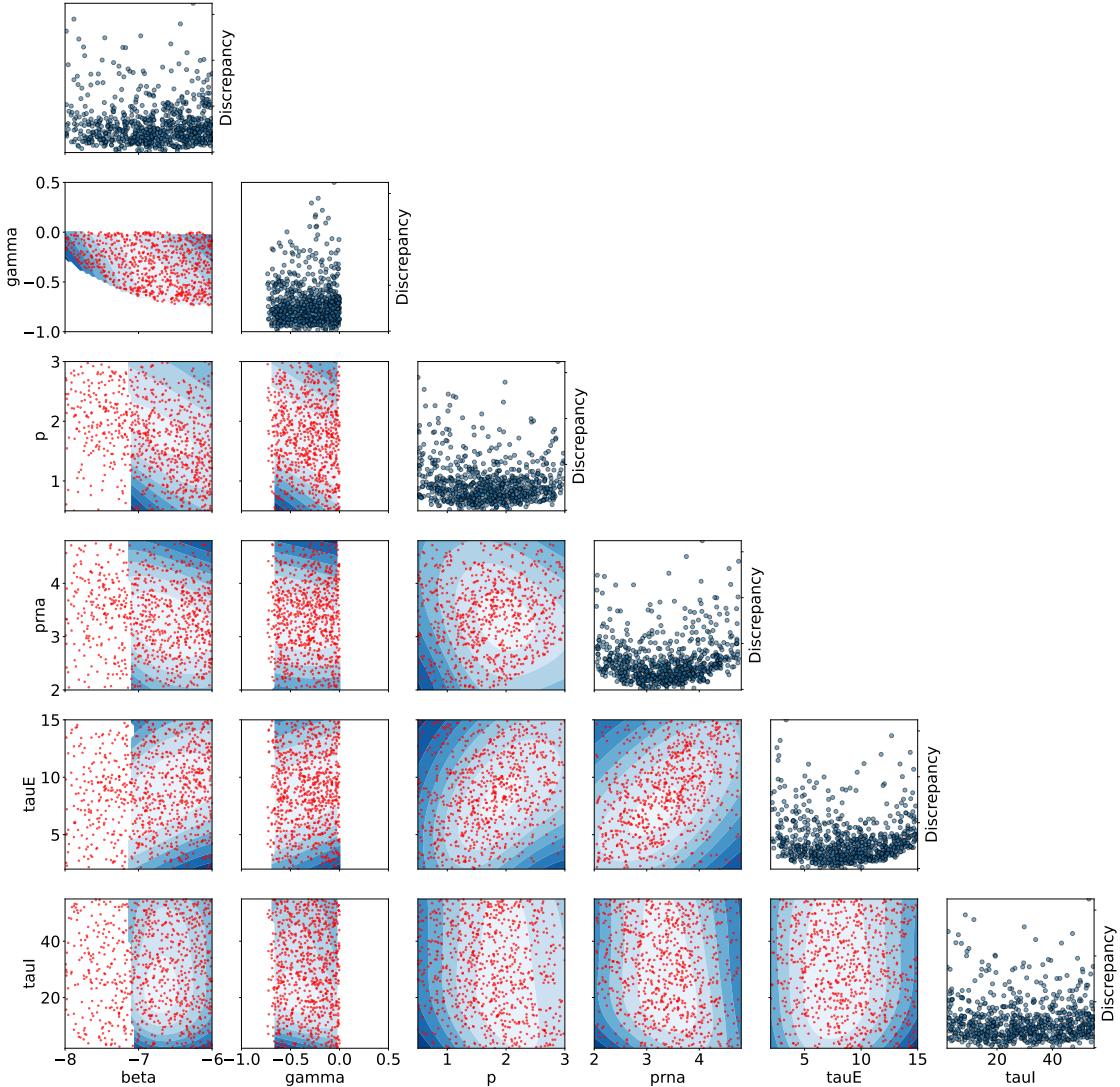


Figure 10: This is a matrix of pairwise plots for Gaussian Process (GP) regression models (Contour plots use a blue color map), illustrating the relationships between parameters and their associated discrepancies. This figure visualize pairwise parameter relationships and their effects on discrepancies, also, provide a detailed visual representation of GP regression predictions and parameter dependencies. The red dots are samples. BOLFI+classifier inference with discrepancy defined in section 3.4 and Figure 8 and learning from experimental data.

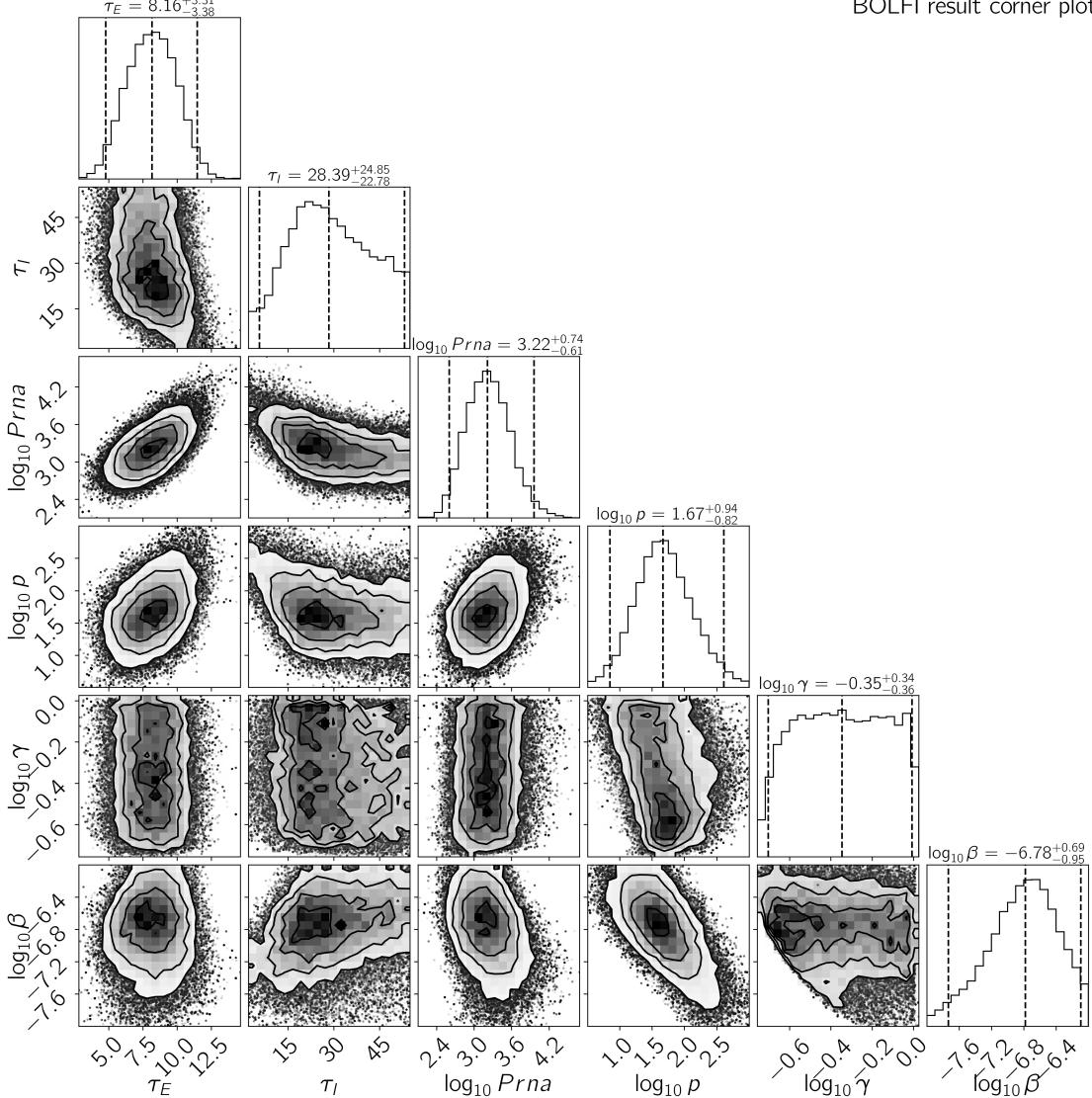


Figure 11: Figure caption, pairwise posterior samples, bolfi euclidean data. Vertical lines are quantiles [0.025, 0.5, 0.975]. The units are  $\tau_E$  (h),  $\tau_I$  (h),  $P_{RNA}$  (VRNA/(cell·h)),  $p$  ([V]/(cell·h)),  $\gamma$  (cell/[V]),  $\beta$  (ml/(cell·h)). This plot is obtained by 100000 samples from BOLFI learned posterior distribution using metropolis sampler. 4 chains of 100000 iterations acquired. Effective sample size and Rhat for each parameter: beta(3039.720542443625; 1.0011963388536813), gamma(10226.27106476491; 1.0007885204262212), p(2636.956305248771; 1.001047925572549), prna(2765.878752279995; 1.0008119623605125), tauE(3772.135815808324; 1.000263512063809), tauI(5259.059487703349; 1.0006640723510034). CPU times: user 25min 29s, sys: 2.27 s, total: 25min 31s.

### 4.3 Interpretation of the results

In Table IV, the estimated rate of infectious virion irreversible entry a cell( $\beta$ ) is  $1.5205 \times 10^{-7}$  [ $1.8450 \times 10^{-8}, 8.0538 \times 10^{-7}$ ] (ml/(cell·h)). This is an extremely small value, indicating that the virus has a low probability of successfully entering a cell. The uncertainty distribution for  $\beta$  in figure 11 is not symmetric; it is slightly skewed to the right. Nonetheless, the distribution's peak is clearly pronounced, indicating a well-defined most probable value.

After an infectious virion enters a cell, the rate of the infectious virion successfully infects the cell (there are possibilities that the virus can enter but can't infect the cell) is estimated as gamma ( $\gamma$ )  $\sim 0.4508[0.1982, 0.9863]$  (cell/[V]) as reported in Table IV. It is around 45%, however, the marginal distribution is rather flat as shown in figure 11 and the uncertainty range is wide. Notably, the lower bound of 0.1982 (cell/[V]) is constrained by the data, while the upper bound approaches the maximum possible value of 1. This indicates that, based on the data, we have only

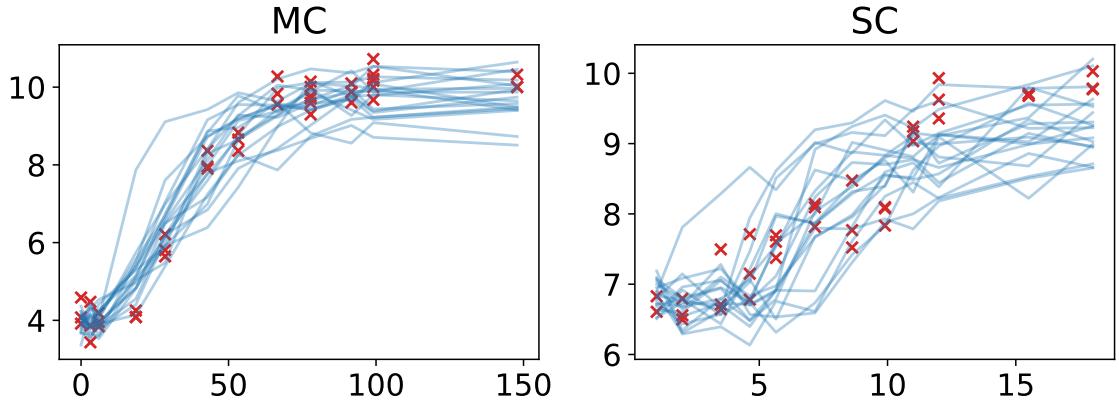


Figure 12: Left Plot (MC): Displays the predicted log-scaled values against the "MC" observed RNA data. Right Plot (SC): Shows the same for "SC" data. The blue lines represent predicted values, while the red crosses indicate observed data points. This figure is a result check to see if the predicted time series can reproduce the data. One can see the prediction can cover the range of data behavior.

established a minimum bound for  $\gamma$ ; no strong preference or information is obtained regarding its most likely value. This is confirmed also in (Quirouette *et al.*, 2024). It is important to note that the flat shape of the marginal distribution on a log scale is primarily a consequence of the uniform prior distribution in log scale.

The production rate for total RNA virions  $P_{RNA}$  is estimated as 1702.1585 (vRNA/(cell·h)) with a 95% credible interval of [401.7908, 9141.1324], as reported in Table IV. This means that one virion RNA produces about 1702.1585 times new total RNA virions per cell per hour. Such a high production rate reflects the rapid growth of the total viral population.

The infectious production rate for infectious virions is estimated as  $P \sim 48.9779[7.0795, 408.3194]$  ([V]/(cell·h)) in Table IV. This means that one infectious virion produces about 48.9779 many new infectious virions per cell per hour during the virus-cell interaction cycle. Comparing to the total RNA virions production rate, this infectious virions production rate is about 34.75 times less. While the population growth speed of infectious virions is significantly less than that of total viral RNA, a production rate of around 49 per hour is still very high. This parameter is a crucial measure of viral dynamics, useful for predicting the speed of virus proliferation and informing medical treatment strategies.

The estimated mean duration for the eclipse phase ( $\tau_E$ ) is 8.142 hours, with a 95% credible interval of [4.781, 11.472](h), as shown in Table IV. According to existing literature, the eclipse phase for influenza A typically ranges from approximately 8 to 10 hours, indicating that our estimate aligns well with known biological data.

The inferred mean duration for the infectious phase tauI ( $\tau_I$ ) is 29.348 hours, with a 95% credible interval of [5.616, 53.241](h), as shown in Table IV. The learned BOLFI target model in figure 17 exhibits a long flat tail indicating that values above the mean are also plausible with similar levels of preference. This behavior is also observed in the synthetic data simulation test shown in figure 9. The infectious period for influenza A in humans is typically about 120 to 168 hours. However, the duration can vary depending on the individual, viral strain, and immune response. Laboratory and clinical studies generally estimate the contagious period (i.e. when the viral shedding occurs) to be around 96 – 168 hours, with some cases shedding virus slightly longer or shorter. Our estimated mean of around 29 hours aligns with the lower part of this range, but the broad tail indicates that the model still considers longer durations possible, especially given variability among individuals. Overall, this suggests that, on average, people may remain infectious for several days, consistent with existing epidemiological data.

#### 4.4 About parameter $\gamma$

Parameter  $\gamma$  represents the probability that an infectious virion, once it has successfully entered a host cell, proceeds to establish a productive infection. Biologically, this accounts for the fact that not all virions entering a cell result in successful infection due to various cellular defense mechanisms, stochastic processes, or structural defects in the virion itself.

In our inference results, the posterior distribution of  $\gamma$  exhibits a broad, relatively flat shape

with a 95% credible interval of [0.1982, 0.9863], and a median estimate around 0.4508 (cell/[V]) (see Table IV and Figure 11). The lower bound of the distribution is well-constrained by the data, suggesting a minimum infectivity threshold. However, the upper bound extends close to the theoretical maximum of 1, with no strong peak within the distribution, indicating that the data provide limited information about the most probable value beyond this lower limit.

This behavior can be partially attributed to the prior distribution. A uniform prior was placed over the logarithmic scale of  $\gamma$ , which, when transformed back to linear scale, inherently gives more weight to smaller values and flattens the posterior unless the data are sufficiently informative. Moreover, the shape of the discrepancy surface with respect to  $\gamma$  is relatively shallow, reflecting a weak sensitivity of the observable data to changes in this parameter within the plausible biological range.

Interestingly, similar behavior is observed in previous work (Quirouette *et al.*, 2024), where marginal posterior distributions for  $\gamma$  also fail to show clear concentration. This convergence across independent inference strategies reinforces the interpretation that the available experimental data contain limited information to tightly constrain this parameter.

From a modeling standpoint, this implies that caution should be taken when interpreting point estimates of  $\gamma$ . The inferred range should be viewed as a conservative estimate indicating that a minimum level of infectivity can be established. These findings highlight the need for improved experimental resolution—such as single-cell infection assays—to reduce uncertainty in this biologically significant parameter.

## 5 Discussions and Conclusion

This study introduces an efficient likelihood-free inference framework for dynamical modeling of viral infection, leveraging the Bayesian Optimization for Likelihood-Free Inference (BOLFI) algorithm. By applying BOLFI to influenza A virus experimental data, we demonstrate that complex mechanistic models with high-dimensional parameter spaces can be inferred accurately and efficiently, even in the absence of explicitly used likelihood functions.

While previous work derived analytical likelihoods for part of the experimental data (Quirouette *et al.*, 2024), our approach deliberately avoids relying on such assumptions, offering a unified and flexible inference strategy applicable to diverse and potentially heterogeneous datasets. Using Gaussian Process (GP) surrogate modeling of the discrepancy between simulated and observed data, coupled with active acquisition of informative simulation points, BOLFI achieves a substantial speed-up over traditional methods. The total runtime was reduced to approximately two hours—21.5 times faster than the MCMC-based baseline—while maintaining high-quality posterior inference.

Our parameter estimates reveal biologically meaningful insights into influenza A dynamics. The low estimated rate of virion entry into host cells ( $\beta$ ) underscores the inefficiency of initial infection events. The wide but bounded uncertainty in the infection probability per entry event ( $\gamma$ ) suggests the data constrain the lower bound of infectivity but lack resolution for a precise peak. High production rates of both total and infectious virions ( $P_{RNA}$ ,  $P$ ) reflect the virus’s rapid replication, with a notable disparity between RNA and infectious output. Estimates for the eclipse ( $\tau_E$ ) and infectious ( $\tau_I$ ) phases are consistent with known biological ranges, with the latter capturing the variability observed in clinical and laboratory studies.

Methodologically, we enhance the BOLFI framework with a GP-based classifier to handle extreme discrepancies and introduce a discrepancy design that integrates multiple types of experimental data. These innovations broaden the applicability of BOLFI to real-world biological inference problems involving heterogeneous data and partial model observability.

In conclusion, our work highlights the practicality, scalability, and interpretability of likelihood-free inference for modeling complex systems. By circumventing the need for analytically tractable likelihoods, even in cases where such formulation exists, we demonstrate that BOLFI can serve as a robust and general-purpose tool for data-driven discovery across diverse fields, including systems biology, ecology, economics, engineering, physics and beyond.

## Acknowledgments

This research was supported by the Finnish Center for Artificial Intelligence (FCAI). The project was initiated at iTHEMS RIKEN while Yingying Xu was supported by the SPDR program at

RIKEN, Japan. Computational resources were provided by the CSC – IT Center for Science, Finland, through access to the Puhti supercomputing infrastructure. We thank Professor Jukka Corander and Professor Catherine Beauchemin for valuable discussions. We are also grateful to Professor Catherine Beauchemin and Dr. Christian Quirouette for explaining the problem and kindly providing the data and simulator from their earlier work.

*Disclosure statement:*

No potential conflict of interest is reported by the authors

## References

- Bussemaker, Jasper H. *et al.*, (2024). “Surrogate-Based Optimization of System Architectures Subject to Hidden Constraints”, *AIAA AVIATION 2024 FORUM*.
- Cresta, D. *et al.*, (2021). “Time to revisit the endpoint dilution assay and to replace the TCID50 as a measure of a virus sample’s infection concentration”, *PLoS Computational Biology*, Vol. 17 No. 10. Erratum in: *PLoS Comput Biol*. 2023 Jan 31;19(1):e1010877. doi: 10.1371/journal.pcbi.1010877, e1009480. DOI: 10.1371/journal.pcbi.1009480.
- Cresta, Daniel *et al.*, (2021). “Time to revisit the endpoint dilution assay and to replace the TCID50 as a measure of a virus sample’s infection concentration”, *PLoS computational biology*, Vol. 17 No. 10, e1009480. DOI: 10.1371/journal.pcbi.1009480.
- El Gammal, Jonas *et al.*, (2023). “Fast and robust Bayesian inference using Gaussian processes with GPry”, *Journal of Cosmology and Astroparticle Physics*, Vol. 2023 No. 10, p. 021. DOI: 10.1088/1475-7516/2023/10/021.
- Gutmann, Michael U and Corander, Jukka (2016). “Bayesian optimization for likelihood-free inference of simulator-based statistical models”, *Journal of Machine Learning Research*, Vol. 17 No. 1, p. 47. available at: <https://jmlr.org/papers/volume17/15-017/15-017.pdf>.
- Ikonomov, Borislav and Gutmann, Michael U. (2020). “Robust Optimisation Monte Carlo”, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. Ed. by Chiappa, Silvia and Calandra, Roberto. Vol. 108. Proceedings of Machine Learning Research. PMLR, pp. 2819–2829. available at: <https://proceedings.mlr.press/v108/ikonomov20a.html>.
- Lintusaari, J *et al.*, (2018). “ELFI: Engine for Likelihood-Free Inference”, *Journal of Machine Learning Research*, Vol. 19 No. 16, pp. 1–7.
- Lintusaari, Jarno *et al.*, (2017). “Fundamentals and Recent Developments in Approximate Bayesian Computation”, *Systematic Biology*, Vol. 66 No. 1, e66–e82. DOI: 10.1093/sysbio/syw077. available at: <https://doi.org/10.1093/sysbio/syw077>.
- Price, Fishman *et al.*, (2017). “Bayesian Synthetic Likelihood”, *Journal of Computational and Graphical Statistics*, Vol. 27 No. 1, pp. 1–11. DOI: 10.1080/10618600.2017.1302882. available at: <https://doi.org/10.1080/10618600.2017.1302882>.
- Quirouette, Christian *et al.*, (2023). “The effect of random virus failure following cell entry on infection outcome and the success of antiviral therapy”, *Scientific Reports*, Vol. 13 No. 1, p. 17243. DOI: 10.1038/s41598-023-44180-w.
- Quirouette, Christian *et al.*, (2024). “Does the random nature of cell-virus interactions during in vitro infections affect TCID50 measurements and parameter estimation by mathematical models?”, arXiv: 2412.12960 [physics.bio-ph]. available at: <https://doi.org/10.48550/arXiv.2412.12960>.
- Rasmussen, Carl Edward and Williams, Christopher KI (2006). *Gaussian processes for machine learning*, MIT press.
- Reed, L. J. and Muench, H. (1938). “A simple method of estimating fifty percent endpoints”, *American Journal of Epidemiology*, Vol. 27 No. 3. <https://doi.org/10.1093/oxfordjournals.aje.a118408>, pp. 493–497.
- Remes, Ulpu (2024). “ELFI Notebooks: Failure-robust BOLFI”, Accessed: 2024-2-20. available at: [https://github.com/uremes/elfi-notebooks/blob/extend\\_model/failure\\_robust\\_BOLFI.ipynb](https://github.com/uremes/elfi-notebooks/blob/extend_model/failure_robust_BOLFI.ipynb).
- Simola, Umberto *et al.*, (2021). “Adaptive Approximate Bayesian Computation Tolerance Selection”, *Bayesian Analysis*, Vol. 16 No. 2, pp. 397–423. DOI: 10.1214/20-BA1211. available at: <https://doi.org/10.1214/20-BA1211>.
- Simon, P. F. *et al.*, (2016). “Avian influenza viruses that cause highly virulent infections in humans exhibit distinct replicative properties in contrast to human H1N1 viruses”, *Scientific Reports*, Vol. 6, p. 24154. DOI: 10.1038/srep24154. available at: <https://www.nature.com/articles/srep24154>.

Srinivas, Niranjan *et al.*, (2010). “Gaussian process optimization in the bandit setting: No regret and experimental design”, *Proceedings of the 27th International Conference on Machine Learning (ICML)*.

## A Simulating the total virion RNA time series data

Ordinary Differential Equation(ODE) is a type of mathematical equation used to describe the dynamics of systems that change continuously over time. In our context, an ODE model is used to describe virus infection dynamics using differential equations that track variables like the number of infected cells and virions over time. The information in this section is summarized from reference (Quirouette *et al.*, 2024).

The ordinary differential equation (ODE) model used herein are given by

$$\frac{dT}{dt} = -\gamma\beta TV/s \quad (9)$$

$$\frac{dE_1}{dt} = \gamma\beta TV/s - \frac{E_1}{\tau_E} \quad (10)$$

$$\frac{dE_i}{dt} = \frac{E_{i-1}}{\tau_E} - \frac{E_i}{\tau_E}, \quad i = 2, 3, \dots, n_E \quad (11)$$

$$\frac{dI_1}{dt} = \frac{E_{n_E}}{\tau_E} - \frac{I_1}{\tau_I} \quad (12)$$

$$\frac{dI_j}{dt} = \frac{I_{j-1}}{\tau_I} - \frac{I_j}{\tau_I}, \quad j = 2, 3, \dots, n_I \quad (13)$$

$$\frac{dV}{dt} = p \sum_{j=1}^{n_I} I_j - cV - \beta TV/s \quad (14)$$

$$\frac{dV_{RNA}}{dt} = p_{RNA} \sum_{j=1}^{n_I} I_j - c_{RNA} V_{RNA} - \beta TV/s \quad (15)$$

The virus infection of a population of  $N_{cells}$  cells bathed in a supernatant of volume  $s$ . In the equations,  $T$  represents the volume of the target cells. Initially, all cells are uninfected, that is  $T(t = 0) = N_{cells}$ , which are exposed to an initial dose of  $V(t = 0) = V_0$  infectious virions. As target cells,  $T$ , encounter infectious virions,  $V$ , in the supernatant of a cell culture, a tissue, or an organ compartment of volume  $s$ , some become infected ( $T \rightarrow E_1$ ).

The rate of successful cell infections by infectious virions is  $\gamma\beta V/s$ , which depends on the concentration of infectious virions  $V/s$ . When a cell becomes infected, it enters ( $T \rightarrow E_1$ ) and traverses ( $E_1 \rightarrow E_2 \rightarrow \dots \rightarrow E_{n_E}$ ) the  $n_E$  compartments of the eclipse phase, during which it is infected, but is not yet producing infectious virions. The infected cell then enters ( $E_{n_E} \rightarrow I_1$ ) and traverses ( $I_1 \rightarrow I_2 \rightarrow \dots \rightarrow I_{n_I}$ ) the infectious phase, during which it produces infectious virions at a constant rate  $p$ . Representing total virions ( $V_{RNA}$ ) corresponding to the number of viral RNA copies as measured by qRT-PCR in units of viral RNA (vRNA). Parameters  $p_{RNA}$  and  $c_{RNA}$  are the corresponding infectious production rate and infectious loss rate for total RNA virions. By definition, the total virions include the infectious virions. Therefore, the condition  $p_{RNA} > p$  must be satisfied.

When an infected cell leaves the last compartment ( $I_{n_I}$ ), it ceases virus production and thus ceases to contribute to the infection kinetics, and possibly undergoes apoptosis. The exponentially distributed durations of the  $n_E$  eclipse (or  $n_I$  infectious) phase compartments together yield an Erlang distributed total duration for the eclipse (or infectious) phase of mean duration  $\tau_E$  (or  $\tau_I$ ), and standard deviation  $\tau_E/\sqrt{n_E}$  (or  $\tau_I/\sqrt{n_I}$ ), where  $n_E$  (or  $n_I$ ) corresponds to the shape parameter of the Erlang distribution. These compartments are not meant to correspond to particular biological states. Rather, they offer a mathematically and computationally expeditious way (compared to delayed, partial, or integro-differential equations) to implement biologically realistic durations for the time spent by infected cells in the eclipse and infectious phases (e.g., normal and lognormal distributions).

Infectious virions (IV) in the supernatant are lost due to loss of infectivity at rate  $c$  or irreversible entry into cells at rate  $\beta/s$ . When one IV is lost to cell entry, it either successfully infects it, resulting in  $\gamma$  infected cell per IV entry, or it does not  $(1 - \gamma)$ . There is a delay of  $\tau_E \pm (\frac{\tau_E}{\sqrt{n_E}})$  hours after a cell infection (eclipse phase) before it begins releasing IV progeny. Thereafter, it releases infectious virions into the supernatant at rate  $\rho$  and total (infectious + non-infectious) virion progeny at rate  $\rho_{RNA}$ . Virus release will persist for  $\tau_I \pm (\frac{\tau_I}{\sqrt{n_I}})$  hours (infectious phase), after which the cell undergoes apoptosis.

## B Simulating the ED assay time series data

A TCID<sub>50</sub> endpoint dilution (ED) assay is usually carried out in a cell culture plate consisting of a number of wells organized into rows and columns. The information in this section is summarized from reference (Quirouette *et al.*, 2024). In our data, we have 8 columns given examined infected number and 4 rows of replicates for each column. Let us consider an example ED experiment in which the first 8 columns of the 36-well plate receive increasing dilutions of the virus sample, all 4 rows within a column receive the same dilution and thus serve as replicates, and the last column is reserved for the sample-free control. Figure 2A in (Quirouette *et al.*, 2023) illustrates one possible random outcome for the number of infectious virions that would be received by each well of this simulated experiment given a virus sample with a known infectious virion (IV) concentration ( $C_{\text{actual}} = 10^6 \text{ IV/ml}$ ), the example dilution factor for each column can be ( $D_1 = 10^{-2}$ ,  $D_2 = 10^{-2.6}$ ,  $D_3 = 10^{-3.2}$ , ...,  $D_{11} = 10^{-8}$ ), and the total volume of inoculum placed in each well ( $V_{\text{inoc}} = 0.1 \text{ ml}$ ). In the data, the dilution factor for each ED assay experiment are given and for some samples in some time steps the dilution factors are varied for the reason to increase the measuring accuracy of the ED assay experiment.

The process of simulating an ED assay outcome. A simulated ED assay experiment for a virus sample with an actual infectious virion concentration  $C_{\text{actual}}$  begins by determining the random number of IV deposited in each well of replicate row  $i$  in dilution column  $j$ ,  $V_0^{i,j}$ . These random numbers are drawn from a Binomial distribution:  $\text{Binomial}(n_j = \frac{V_{\text{inoc}} D_j}{V_{\text{vir}}}, p = C_{\text{actual}} V_{\text{vir}})$ , where  $V_{\text{inoc}}$  is the total volume of inoculum placed in each well,  $D_j \in (0, 1]$  is the dilution factor for column  $j$ , and  $V_{\text{vir}} = 5.236 \times 10^{-16} \text{ ml}$  is the volume of a single influenza A virion ((Daniel Cresta *et al.*, 2021)). To determine whether any one well becomes infected, a uniform random number  $r \in [0, 1]$  is drawn, and the well is uninfected if  $r < (P_{V \rightarrow \text{Extinction}})^{V_0^{i,j}}$ , and infected otherwise. Here,  $P_{V \rightarrow \text{Extinction}}$  is the probability that a single infectious virion will become extinct and the expression is

$$P_{V \rightarrow \text{Extinction}} = \left[ 1 - \frac{\gamma}{1 + c/(\beta N_{\text{cells}}/s)} \right] + \frac{\gamma}{1 + c/(\beta N_{\text{cells}}/s)} \left[ \frac{\mathcal{B}(1 - P_{V \rightarrow \text{Extinction}})}{\eta_I + 1} + 1 \right]^{-n_I}, \quad (16)$$

where  $\mathcal{B} = p\tau_I$  is the burst size. This can also be expressed as  $P_{V \rightarrow \text{Extinction}} = 1 - P_{V \rightarrow \text{Establishment}}$ , where for  $n_I = 1$ , the expression for the establishment probability reduces to

$$P_{V \rightarrow \text{Establishment}} = \frac{R_0 - 1}{\mathcal{B}} = \frac{\gamma}{1 + c/(\beta N_{\text{cells}}/s)} - \frac{1}{p\tau_I}, \quad (17)$$

where  $R_0$  is the average basic reproductive number (Quirouette *et al.*, 2023). In other studies, the most likely  $\log_{10}$  infectious dose concentration ( $\log_{10}(\text{SIN}/\text{ml})$ ) is determined by providing the number of infected wells in each dilution column to the midSIN calculator (Daniel Cresta *et al.*, 2021). However, in this study, we use the ED assay outcome data directly. Therefore, we do not use this step to calculate the estimated SIN.

## C Data

Figure 13: Data table of sH1N1 SC RNA data.(Simon *et al.*, 2016)

Time post infection [h]	Virus [vRNA/ml]
1	$4.05 \times 10^6$
1	$6.73 \times 10^6$
2	$3.14 \times 10^6$
2	$3.61 \times 10^6$
2	$6.22 \times 10^6$
3.5	$3.13 \times 10^7$
3.5	$5.12 \times 10^6$
3.5	$4.39 \times 10^6$
4.63	$6.03 \times 10^6$
4.63	$1.41 \times 10^7$
4.63	$5.14 \times 10^7$
5.65	$4.96 \times 10^7$
5.65	$2.37 \times 10^7$
5.65	$4.02 \times 10^7$
7.17	$1.38 \times 10^8$
7.17	$6.54 \times 10^7$
7.17	$1.24 \times 10^8$
8.62	$2.98 \times 10^8$
8.62	$5.87 \times 10^7$
8.62	$3.33 \times 10^7$
9.9	$6.81 \times 10^7$
9.9	$1.20 \times 10^8$
9.9	$1.25 \times 10^8$
11	$1.08 \times 10^9$
11	$1.49 \times 10^9$
11	$1.74 \times 10^9$
12	$8.53 \times 10^9$
12	$4.22 \times 10^9$
12	$2.28 \times 10^9$
15.5	$5.14 \times 10^9$
15.5	$4.78 \times 10^9$
15.5	$4.83 \times 10^9$
18	$5.92 \times 10^9$
18	$1.07 \times 10^{10}$
18	$6.06 \times 10^9$

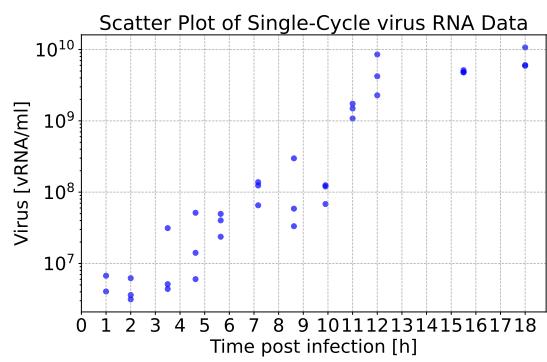


Figure 14: sH1N1 Single Cycle virus RNA data scatter plot. Y axes is log 10 scale. Figure 14 is a visualization of data in figure 13. Data is from (Simon *et al.*, 2016).

Figure 15: Data table of sH1N1 MC RNA data.(Simon *et al.*, 2016)

Time post infection [h]	Virus [vRNA/ml]
0.0	$3.86 \times 10^4$
0.0	$1.19 \times 10^4$
0.0	$8.22 \times 10^3$
3.166667	$7.25 \times 10^3$
3.166667	$2.72 \times 10^3$
3.166667	$2.99 \times 10^4$
6.0	$6.99 \times 10^3$
6.0	$1.54 \times 10^4$
6.0	$9.04 \times 10^3$
18.58333	$1.79 \times 10^4$
18.58333	$1.21 \times 10^4$
18.58333	$1.22 \times 10^4$
28.5	$4.42 \times 10^5$
28.5	$6.43 \times 10^5$
28.5	$1.63 \times 10^6$
42.86666	$2.30 \times 10^8$
42.86666	$7.97 \times 10^7$
42.86666	$9.04 \times 10^7$
53.3	$4.47 \times 10^8$
53.3	$6.77 \times 10^8$
53.3	$2.30 \times 10^8$
66.58334	$6.82 \times 10^9$
66.58334	$1.88 \times 10^{10}$
66.58334	$3.57 \times 10^9$
77.83334	$5.88 \times 10^9$
77.83334	$1.39 \times 10^{10}$
77.83334	$1.02 \times 10^{10}$
77.83334	$1.98 \times 10^9$
77.83334	$5.09 \times 10^9$
77.83334	$3.92 \times 10^9$
91.66666	$7.56 \times 10^9$
91.66666	$1.25 \times 10^{10}$
91.66666	$4.02 \times 10^9$
99.08334	$1.01 \times 10^{10}$
99.08334	$1.46 \times 10^{10}$
99.08334	$4.72 \times 10^9$
99.08334	$2.10 \times 10^{10}$
99.08334	$5.27 \times 10^{10}$
99.08334	$1.63 \times 10^{10}$
147.8333	$1.02 \times 10^{10}$
147.8333	$2.09 \times 10^{10}$
147.8333	$1.00 \times 10^{10}$

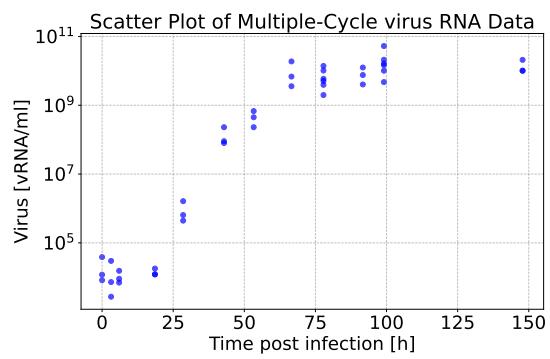


Figure 16: sH1N1 Multiple Cycle virus RNA data scatter plot. Y axes is log 10 scale. Figure 16 is a visualization of data in figure 15. Data is from (Simon *et al.*, 2016).

Table V: Multiple Cycle ED Data Table(Simon *et al.*, 2016)

Time post infection [h]	The dilution factor [ $10^n$ ]	ED assay infection counts
0	-1 -2 -3 -4 -5 -6 -7 -8	0 0 0 0 0 0 0 0
0	-1 -2 -3 -4 -5 -6 -7 -8	1 0 0 0 0 0 0 0
0	-1 -2 -3 -4 -5 -6 -7 -8	0 0 0 0 0 0 0 0
3.16667	-1 -2 -3 -4 -5 -6 -7 -8	2 0 0 0 0 0 0 0
3.16667	-1 -2 -3 -4 -5 -6 -7 -8	1 0 0 0 0 0 0 0
3.16667	-1 -2 -3 -4 -5 -6 -7 -8	0 0 0 0 0 0 0 0
6	-1 -2 -3 -4 -5 -6 -7 -8	0 0 0 0 0 0 0 0
6	-1 -2 -3 -4 -5 -6 -7 -8	0 0 0 0 0 0 0 0
6	-1 -2 -3 -4 -5 -6 -7 -8	1 0 0 0 0 0 0 0
18.5833	-1 -2 -3 -4 -5 -6 -7 -8	4 1 0 0 0 0 0 0
18.5833	-1 -2 -3 -4 -5 -6 -7 -8	4 1 0 0 0 0 0 0
18.5833	-1 -2 -3 -4 -5 -6 -7 -8	4 2 0 0 0 0 0 0
28.5	-3 -4 -5 -6 -7 -8 -9 -10	3 1 0 0 0 0 0 0
28.5	-3 -4 -5 -6 -7 -8 -9 -10	3 1 0 0 0 0 0 0
28.5	-3 -4 -5 -6 -7 -8 -9 -10	2 0 0 0 0 0 0 0
42.8667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 1 0 0 0
42.8667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 0 0
42.8667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 0 0 0 0
53.3	-4 -5 -6 -7 -8 -9 -10 -11	4 4 2 0 0 0 0 0
53.3	-4 -5 -6 -7 -8 -9 -10 -11	4 4 1 0 0 0 0 0
53.3	-4 -5 -6 -7 -8 -9 -10 -11	4 4 1 0 0 0 0 0
66.5833	-4 -5 -6 -7 -8 -9 -10 -11	4 4 2 0 0 0 0 0
66.5833	-4 -5 -6 -7 -8 -9 -10 -11	4 4 3 0 0 0 0 0
66.5833	-4 -5 -6 -7 -8 -9 -10 -11	4 4 2 0 0 0 0 0
77.8333	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 3 0 0
77.8333	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 1 0
77.8333	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
91.6667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
91.6667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
91.6667	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 3 1 0 0
99.0833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 3 1 0 0
99.0833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 2 0 0
99.0833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 0 0
147.833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 2 1 0 0
147.833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 1 0 0 0
147.833	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 1 0 0 0

Table VI: Single Cycle ED Data Table(Simon *et al.*, 2016)

Time post infection [h]	The dilution factor [ $10^n$ ]	ED assay infection counts
1	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 0 0 0 0
1	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 3 1 0 0
1	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 2 0 0 0
2	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 1 0 0 0
2	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 2 0 0 0
2	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 0 0 0 0
3.5	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 0 0 0 0
3.5	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
3.5	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 1 0 0 0
4.63	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 2 0 0 0
4.63	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 0 0 0 0
4.63	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 2 0 0 0
5.65	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 2 0 0
5.65	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
7.17	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 0 0 0
7.17	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 3 1 0 0
7.17	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 0 0
8.62	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 0 0
8.62	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 3 1 0 0
8.62	-1 -2 -3 -4 -5 -6 -7 -8	4 4 4 4 4 1 0 0
9.9	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 0 0 0 0
9.9	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 1 0 0 0
9.9	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 3 0 0 0
11	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 2 0 0 0
11	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 1 0 0 0
11	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 3 1 0 0 0
12	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 2 0 0 0
12	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 1 0 0 0
12	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 2 0 0 0
15.5	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 3 0 0 0
15.5	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 3 0 0 0
15.5	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 1 0 0 0
18	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 1 0 0 0
18	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 3 0 0 0
18	-2 -3 -4 -5 -6 -7 -8 -9	4 4 4 4 3 1 0 0

## D Additional figures

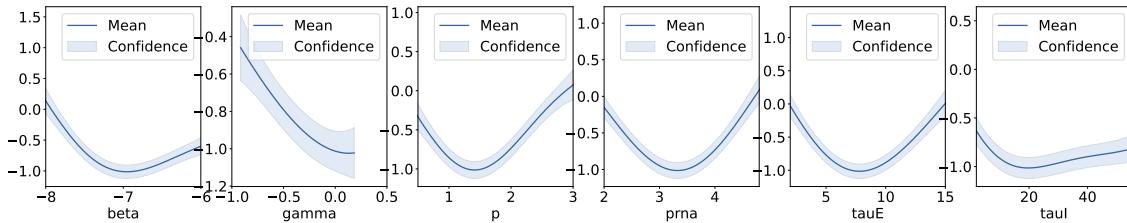


Figure 17: Plot of one dimensional slices of the learned multidimensional GP surrogate model(the target model in BOLFI) for each parameter dimension. The solid lines represent mean value and shadow represents the corresponding confidence of the estimate. The minimum of the lines corresponding the optimal estimate of the parameter.

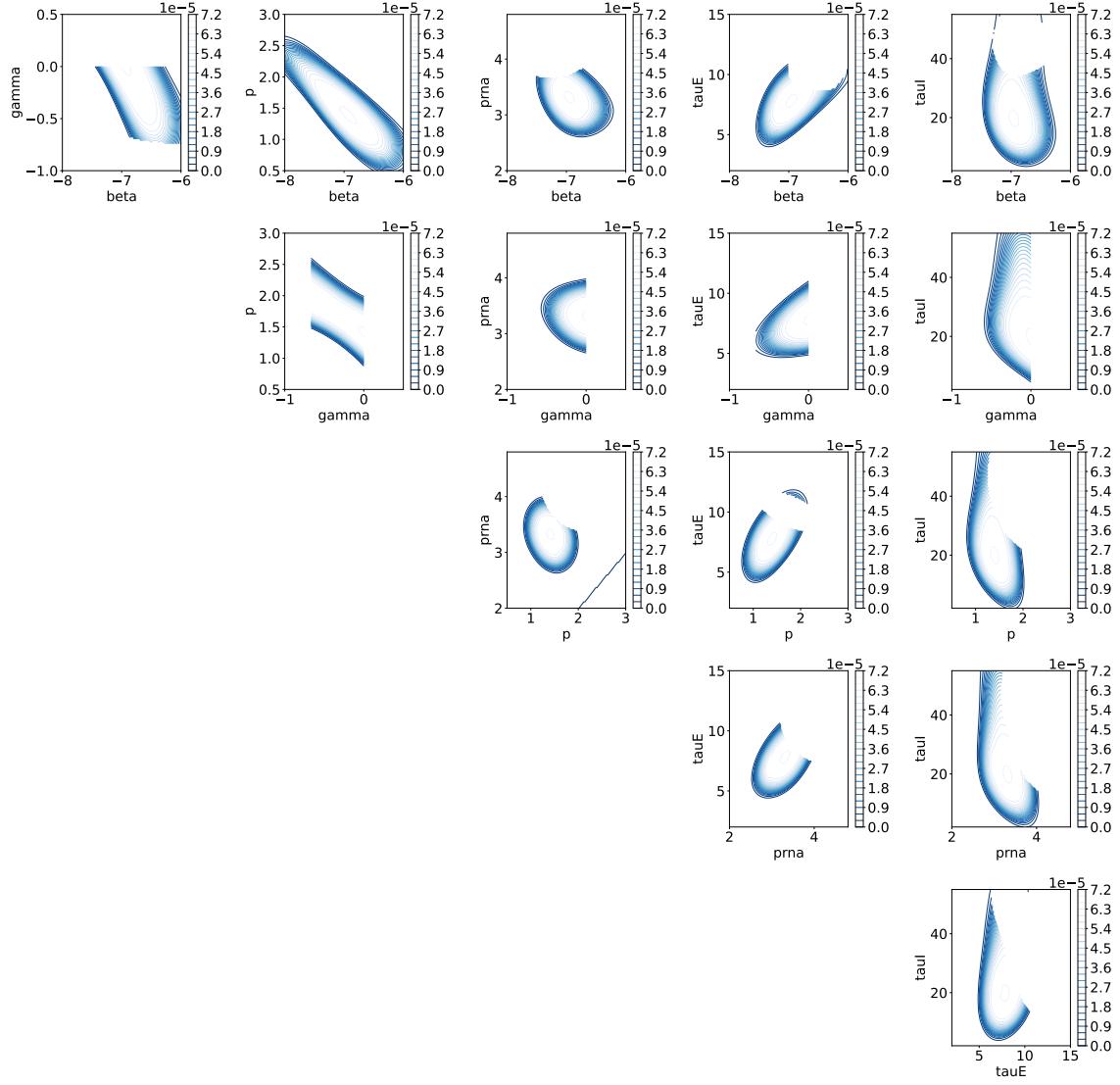


Figure 18: Two dimensional slices of posterior probability density function for each pair of parameters. The cut area are removed by the classifier, because the simulator fails(parameter values are not valid) in those area.

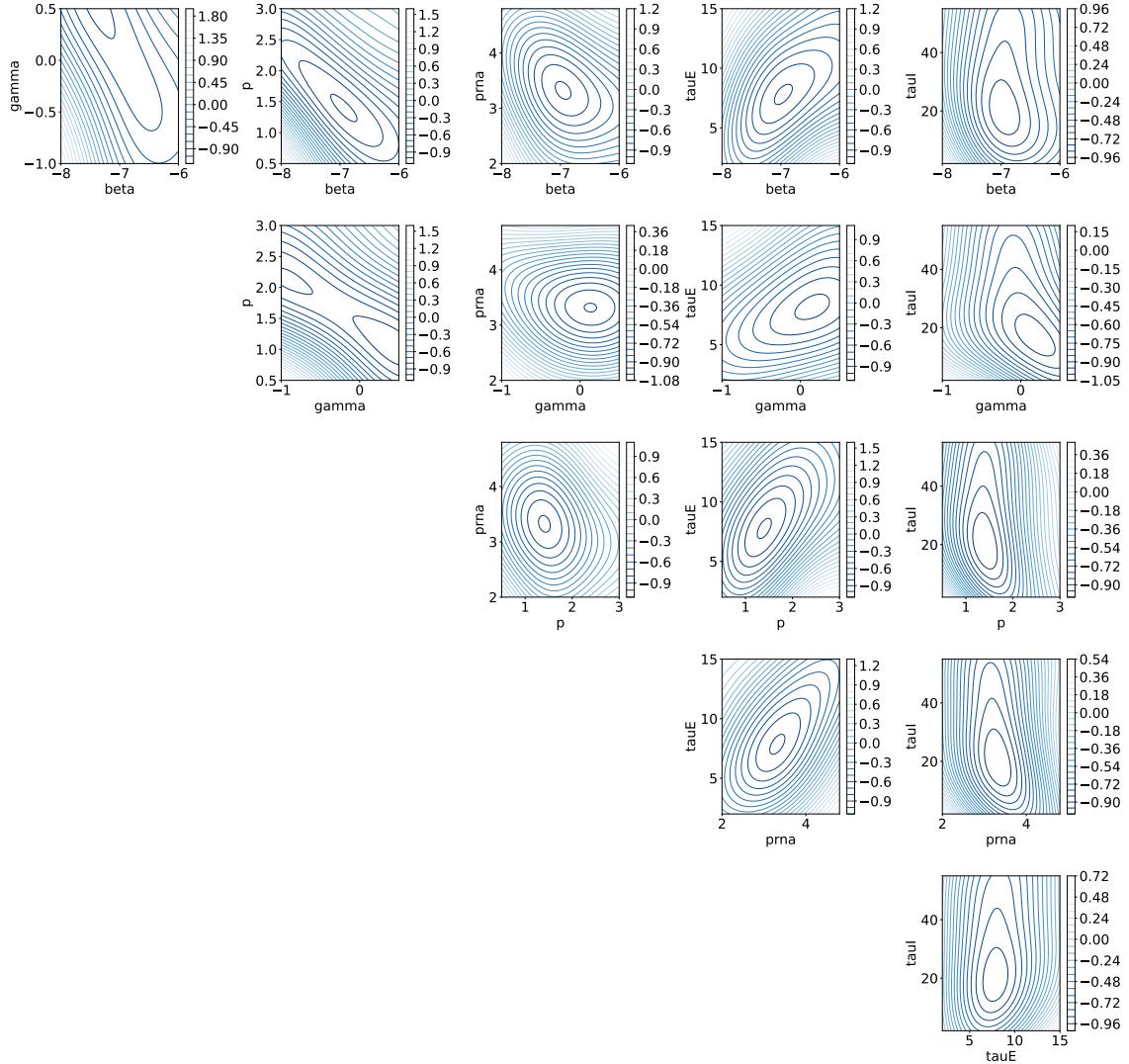


Figure 19: This figure generates 2 dimensional contour plots of the GP predicted function across all pairs of dimensions in a multidimensional parameter space. It emphasizes pairwise interactions in a high-dimensional parameter space.

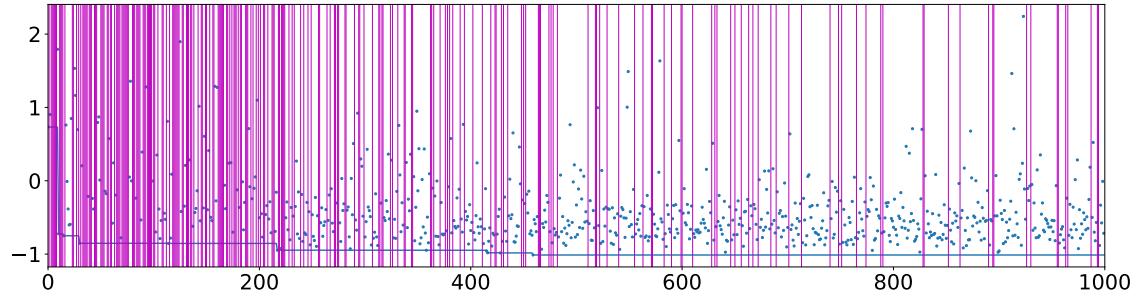


Figure 20: This figure visualizes the progress of the BOLFI algorithm by plotting the observed minimum discrepancies over iterations. This is showing how the discrepancy reduces over iterations. This visualizes the optimization process over time. Magenta color vertical lines highlight non-finite discrepancies which correspond to cases when the simulator fails.