

Final Report for MEng Project

Parameter Optimisation

for Invasive Aspergillosis *in silico* Model

Student: Jiujia Zhang
CID: 01203777

Supervisors: Dr Reiko Tanaka
Miss Tara Hameed

Department: Department of Bioengineering
Project Dates: Oct. 2019 - Jun. 2020

Date: June 16, 2020
Word Count: 5763

Submitted in partial fulfilment of the requirements for the award of MEng in Biomedical Engineering from Imperial College London

Acknowledgements

I would like to express my heartfelt gratitude to all the special people during my undergraduate degree at Imperial College London.

To my project supervisors **Dr Reiko Tanaka** and **Miss Tara Hameed** who are the most influential mentors in shaping my future academic path, for their generous support and valuable guidance throughout the project.

To my personal tutor **Professor Mengxing Tang** for all the advice in both academic and personal matters.

To my violin teacher **Mrs Esther Smith** for providing the best musical education which has always refreshed my mind from work overload.

To my friends: Junning, Ocean, Shulin, Tianruo and Yu, for all the laughter which has brightened up my life.

To my parents and family for the unconditional support and love, as always.

To Nikolaos for the company whenever I was weak or strong, whenever I was up or down.

Abstract

With the recent advances in computation, *in silico* modelling has rapidly expanded its application in accelerating therapeutic strategies discovery in immunology. Consequently, the development of parameter estimation methodologies has received extensive attention in order to facilitate the validation of *in silico* models. The experimental time course measurement for parameter estimation often suffers from the features of sparse, irregular time interval and unpredictable noise levels in immunology. However the effects of such data features to the applicability of parameter estimation algorithms are under-reported. This project aims at understanding the attainability of classical parameter estimation algorithms under sparse and latent data set up. This was achieved by conducting simulation studies using an invasive aspergillosis model where parameter estimation with sparse experimental data was previously attempted. Overall, the results suggest that parameter value retrieval for the testing model was not possible with sparse and latent synthetic data using either genetic algorithm or Gauss-Newton iteration. In addition, it was found that the unsuccessful parameter retrieval was not solely ascribed to the data sparsity, but also dominated by model structure and model stability. To conclude, model identifiability, which is principally influenced by model structure and data sparsity, and model stability should be examined prior to exhaustively attempting parameter estimation.

Contents

1	Introduction	1
2	Project Specification	2
2.1	Invasive Aspergillosis	2
2.2	<i>In silico</i> Model	3
2.3	Testing Model and Challenges	3
3	Methodology	5
3.1	Synthetic Data	5
3.2	Parameter Estimation Algorithms	6
3.2.1	Genetic Algorithm	7
3.2.2	Gauss-Newton Iteration	10
3.3	ODE Solving	12
4	Results	14
4.1	Genetic Algorithm	14
4.1.1	Parameter Retrieval	14
4.1.2	Local Minimum	15
4.1.3	Curve Fitting	16
4.1.4	Parameter Non-identifiability	17
4.2	Gauss-Newton Iteration	19
4.2.1	Parameter Retrieval	19
4.2.2	Model Instability	20
5	Discussion	21
6	Conclusion	23
	References	24
	Appendix A: Algorithm Implementation	31
	Appendix B: Supplementary for Genetic Algorithm Results	32

1 Introduction

Parameter estimation or model calibration is recognised as a curve fitting problem, it identifies parameter numerical values that reduce the discrepancy between model simulations and experimental data sets. Hence, the model simulations reveal dynamical behaviours of the experimental observations [1]. In particular, parameter estimation is an essential process in completing an *in silico* model in systems biology, since the kinetic rates between reactions are difficult to measure in laboratories and therefore are usually not readily available [2]. Moreover, parameter estimation plays an important role in the reliability of model predictive behaviours, which is critical in improving our current understanding in immunology [2].

Optimisation methods have contributed significantly in the application of parameter estimation for *in silico* models [3]. With the recent advancement in computations and rapidly expanded modelling applications in the area of systems biology, various parameter estimation algorithms have been developed in the past few decades, which can be categorised into stochastic global optimisation [4] and gradient-based local optimisation algorithm [5]. Genetic algorithm [6] - a popular stochastic global optimisation algorithm - has been extensively used for parameter estimation in modelling bioprocesses [7–9]. For example, the successful applications of genetic algorithm include: cell signalling transduction pathways [10–12], lymphocyte recirculation [13], breast cancer [14], subtilin production [15], pharmacokinetics [16], myocardium biomechanical properties [17] and HIV [18]. Moreover, there are several variations of genetic algorithm for combating problems resultant from different types of models[19–22]. On the other hand, Gauss-Newton algorithm - a well known local optimiser - has also been improved for systems of differential equation model calibration applications [23–30]. Gauss-Newton iteration based algorithms have also been utilised in estimating kinetic parameters for studying biological systems, such as, bioreactions in diabetes [31, 32], hepatic transportation[33, 34], enzyme kinetics [35] and tumor metabolism [36].

However, limited experimental data for parameter estimation especially in computational immunology [2] have called into questions the applicability of these parameter estimation algorithms, since the sparse data set up was not addressed in either the proposed parameter estimation frameworks, nor in the successful applications. Therefore, the aim of this project is to understand whether parameter estimation is achievable given that there is limited amount of experimental data available. The objectives of this project are: (i) generating synthetic data that mimics the degree of sparsity and latency of experimental data which was previously encountered in invasive aspergillosis modelling, (ii) conducting simulation studies on parameter retrieval that uses both genetic algorithm and Gauss-Newton algorithm with synthetic data. Moreover, issues arose during the simulation study by using classical parameter estimation techniques with sparse and latent data were identified. Subsequently, the potential solutions for sparse data model calibration were proposed in order to combat problems encountered in this simulation study.

2 Project Specification

This study was motivated by the demanding parameter estimation applications in the field of computational immunology, where limited data availability challenges the applicability of the contemporary parameter estimation techniques. For this purposes, we chose an invasive aspergillosis model which was attempted to be calibrated with sparse experimental data collected from different literature as the testing model. In this section, the inspiration from both the biological background and the *in silico* modelling aspect of invasive aspergillosis will be introduced. The testing model, available data sparsity and issues with parameter estimation from previous attempt will also be presented.

2.1 Invasive Aspergillosis

Invasive aspergillosis (IA) is a type of pulmonary fungal infection that is commonly caused by an opportunistic fungal species *Aspergillus fumigatus* [37] through daily inhalations [38] as illustrated in figure 1. Such infection is particularly life-threatening for immuno-compromised patients [39] with an unacceptably high mortality rate of up to 90% [37]. Moreover, the survival rate has not even been marginally improved as the advancement in medical equipment at hospitals [40], which could be attributed in the increased resistance to anti-fungal drugs from several fungal species in the past two decades [41–43].

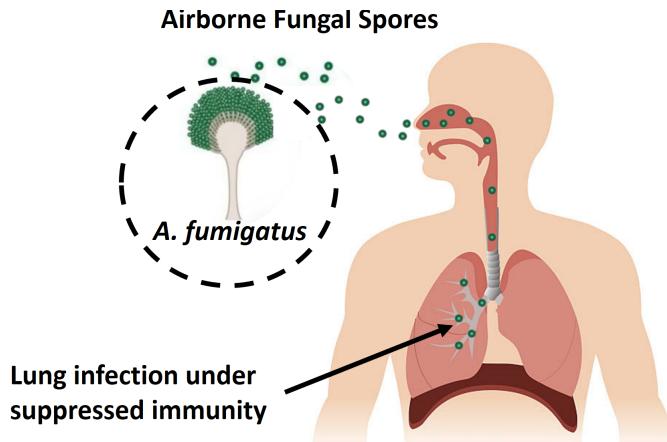


Figure 1: The airborne fungal spores which are produced by ubiquitous *A. fumigatus* arrive the respiratory system through daily inhalation. It causes pulmonary infection when the host immunity is weakened.

The morbidity and mortality caused by IA is still not under control, despite it has been extensively studied experimentally and clinically [40]. Therefore, it is urgent to seek for novel IA infection treatments. Currently, immunotherapy that boosts the host immune response by administering cytokines, is proposed as a potential strategy, since the IA infectious outcome is closely related to host immunity status [44]. However, the incomplete understanding of such treatment and limited number of clinical trials impede the development of immunotherapy in clinical usage.

2.2 *In silico* Model

In silico models are computational programs that can be simulated to reproduce the observed behaviour of complex systems. The current state-of-art models in immunology are mainly based on mathematical description and agent-based modelling [2]. With the significant advancement in computations, *in silico* models have been seen as an indispensable tool in understanding pathogenesis and in drug development [45] where information cannot be obtained from experiments due to technology limitations or ethical considerations [46].

In the past twelve years, *in silico* models have been utilised in studying IA, hence contributed in accelerating the IA therapeutic strategies discovery. Some examples include: the influence of genetic variations in IA pathogenesis [47], the potential anti-fungal targets in treating IA [48–50], the regulatory pathways at endothelial cells when in contact with *A. fumigatus* [51], the fungal spores clearance mechanisms in lungs [52] and the role of iron in host-pathogen interactions [53].

2.3 Testing Model and Challenges

Our testing model (referred to as *fungus model*) is an extended version of a previously validated IA model [52] by T. Hameed, which models the cytokine dynamics in innate immune system using a set of four ordinary differential equations (ODE) as:

$$\begin{aligned} \frac{d[F]}{dt} &= \phi[F] - k_{NF}[N][F] \\ \frac{d[C_1]}{dt} &= \tilde{k}_C[F] - \delta_{C_1}[C_1] \\ \frac{d[C_2]}{dt} &= k_h[F] - \delta_{C_2}[C_2] \\ \frac{d[N]}{dt} &= \tilde{\alpha}_1[C_1] + \tilde{\alpha}_2[C_2] - k_{NF}[N][F] - \delta_N[N] \end{aligned} \quad (1)$$

where $[F]$, $[C_1]$, $[C_2]$ and $[N]$ represent fungal burden [10^6 Cells], TNF- α concentration [ng ml^{-1}], IL-8 concentration [ng ml^{-1}] and Neutrophil [10^6 Cells], respectively. ϕ , k_{NF} , \tilde{k}_c , δ_{C1} , k_h , δ_{C2} , $\tilde{\alpha}_1$, $\tilde{\alpha}_2$ and δ_N are 9 parameters that represents the reaction rates, and their numerical values need to be estimated.

Previously, parameter estimation for the fungal model (equation 1) has been attempted by T. Hameed with experimental data collected from literature [54–60] as shown in figure 2, and the estimated numerical values for those 9 parameters in fungal model (ϕ , k_{NF} , \tilde{k}_c , δ_{C1} , k_h , δ_{C2} , $\tilde{\alpha}_1$, $\tilde{\alpha}_2$ and δ_N) are shown in table 1. However, using the estimated parameter values shown in table 1, the model simulation does not match with experimental data under different initial conditions. Therefore, these 9 free parameters in the model (equation 1) are still undefined.

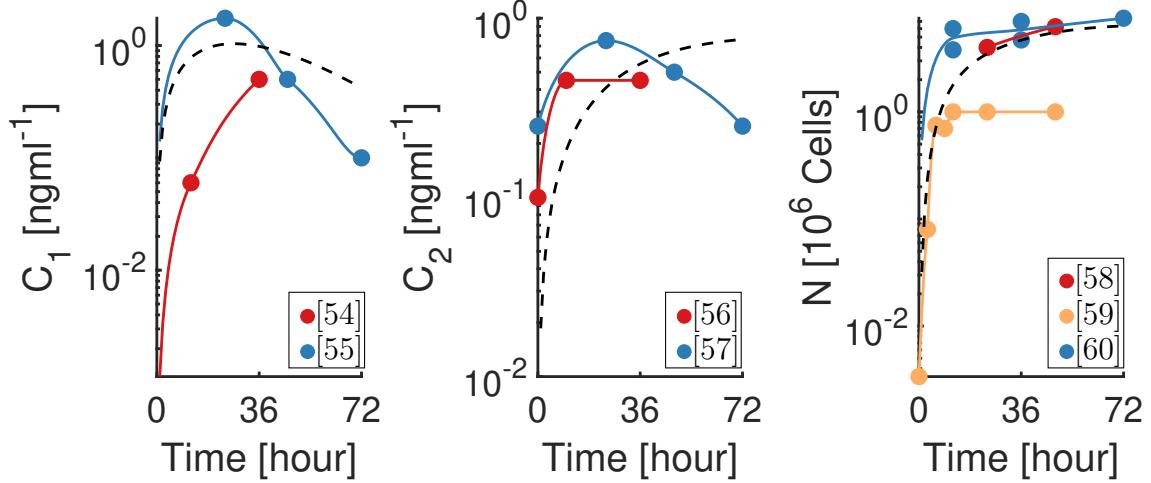


Figure 2: Available experimental data for model parameter estimation (shown in dots) and model simulation with previously estimated parameters shown in table 1 (dashed lines), where C_1 are collected from: [54] (●) and [55] (○), C_2 are from [56] (●) and [57] (○), data for N are from [58] (●), [59] (○) and [60] (○). All the data are with initial inoculum 3×10^7 from intratracheal infected homogenate mice (if infected routine and mice sample information is available in the literature). The solid lines with corresponding colours with the filled circles are the cubic polynomial fitting for data points collected from the corresponding literature.

Table 1: Previous estimated parameters θ by T.Hameed using experimental data shown in figure 2.

θ	ϕ	k_{NF}	\tilde{k}_c	δ_{C1}	k_h	δ_{C2}	$\tilde{\alpha}_1$	$\tilde{\alpha}_2$	δ_N
Value	1.4×10^{-8}	6.4×10^{-3}	3.2×10^{-3}	6.6×10^{-2}	6.2×10^{-4}	3.4×10^{-4}	6.6×10^{-1}	4.0×10^{-1}	6.1×10^{-2}

* units and physical meaning of parameters are omitted since the focus of the project is the numerical value of these parameters.

Parameter estimation for fungal model is challenging due to the following reasons: (i) there are no measurements for fungal burden [F] which is defined as a *latent state variable*, (ii) the observations for $[C_1]$, $[C_2]$ and $[N]$ are sparse and (iii) the measurements are collected from different experiments as shown in figure 2, which are possibly subject to different amount of noise. These commonly existing data features in computational immunology are rarely addressed in the applications of parameter estimation. Hence, optimal selection of parameter estimation paradigm for fungal model remains unclear. Therefore, the fungal model will be used in this project in order to address the problems arising from classical parameter estimation techniques.

3 Methodology

The overview of the simulation study is summarised by the flow chart in figure 3: parameter estimation is performed using sparse *synthetic data*, that is the data generated from a model with known parameter values. Hence, the estimated parameter values $\hat{\theta}$ and the model simulation $f(\mathbf{x}(t), \hat{\theta})$ can be compared with the nominal parameter values θ and synthetic data. The remaining section is organised as following: (i) introducing the synthetic data used for this study, (ii) formulating parameter estimation algorithms - genetic algorithm and Gauss-Newton algorithm, (iii) computation set up regarding to ODE solving and the relevant modifications to each parameter estimation algorithms. The algorithms output $\hat{\theta}$ and the model simulation $f(\mathbf{x}(t), \hat{\theta})$ will be presented and analysed in section 4 and section 5, respectively.

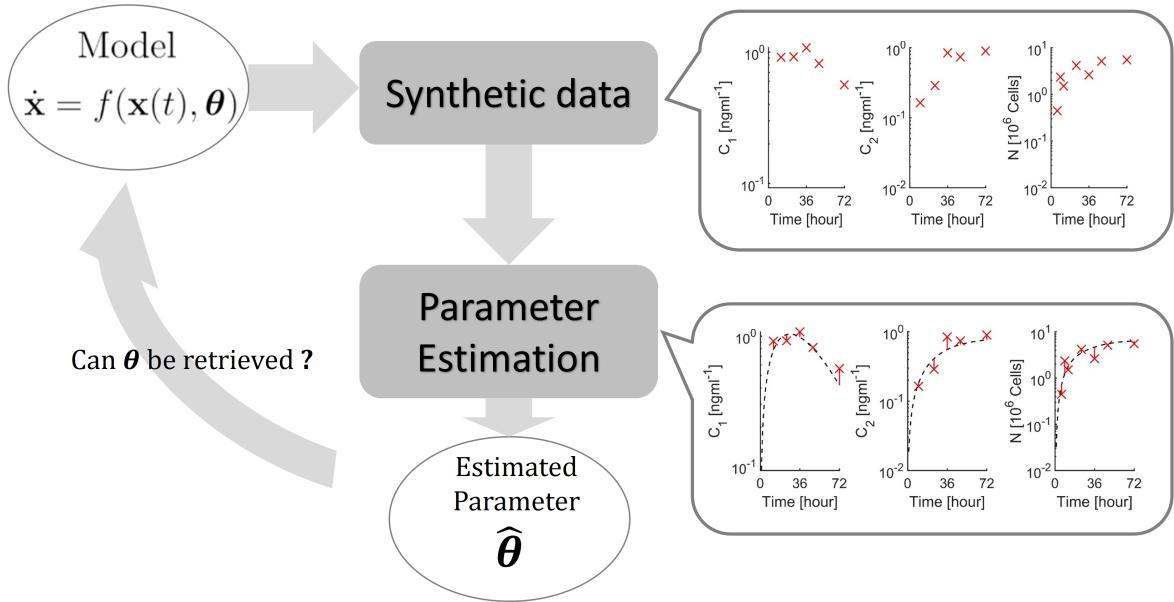


Figure 3: Overview of the simulation study in order to understand whether parameter estimation using classical parameter estimation techniques is achievable given sparse synthetic data.

3.1 Synthetic Data

The synthetic data was designed to mimic the features of experimental data used for fungal model calibration from previous attempt (figure 2). Hence, problems arose in sparse data parameter estimation can potentially be recognised through simulation studies.

The synthetic data (figure 4) are subsampled from the numerical solution of the fungal model (equation 1) at the time points where experimental measurements were made (figure 2). Therefore, the features of latency, sparsity and non-simultaneously measured state variables from previously encountered experimental data for parameter estimation is preserved. The synthetic data for 4 state variables $[F]$, $[C_1]$, $[C_2]$ and $[N]$ are at $t = [0]$, $t = [0, 12, 24, 36, 48, 72]$, $t = [0, 10, 24, 36, 48, 72]$ and $t = [0, 3, 6, 9, 12, 24, 36, 48, 72]$, respectively.

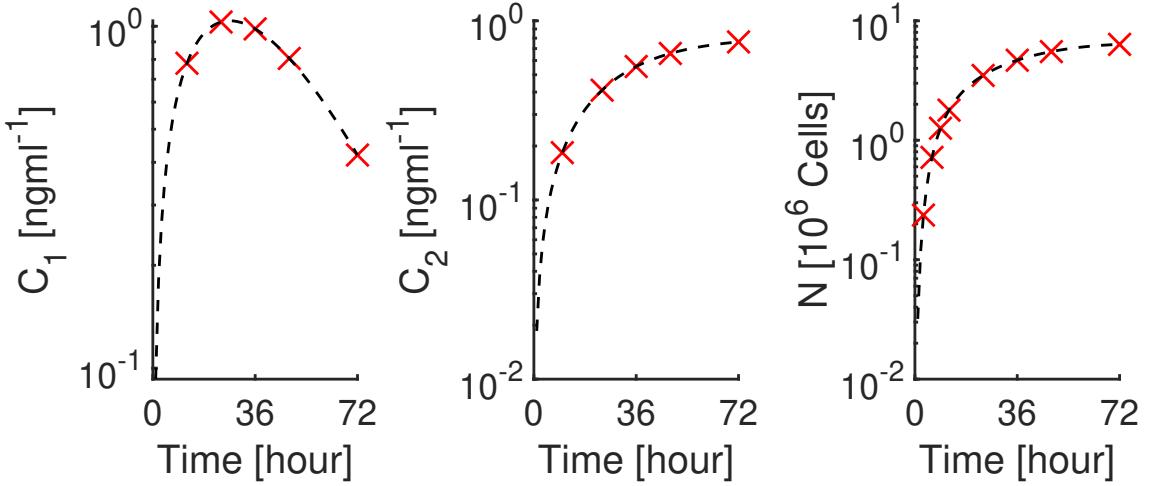


Figure 4: Synthetic data (\times) used for simulation study that mimics the features of previously encountered experimental data in modelling invasive aspergillosis (figure 2). Numerical solution (- -) of the fungal model (equation 1) is solved by MATLAB *ode45()* with $t = [0 : 1 : 72]$, and initial condition $[F, C_1, C_2, N] = [3 \times 10^7, 0, 0, 0]$ which is the same as the previously used experimental data (figure 2). Then numerical solutions are sub-sampled at the time points when experimental measurements were made to form a set of synthetic data.

3.2 Parameter Estimation Algorithms

The purpose of parameter estimation is to match the model simulation with experimental measurements, hence mathematical optimisation is often used to search for the optimal parameter values. In order to explain the formulation of general parameter estimation algorithms for ODE models, we will define a generalised ODE model as:

$$\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\theta}); \quad \mathbf{x}(t_0) = \mathbf{x}_0 \quad (2)$$

where \mathbf{x} is an n -dimensional vector of state variables, t is a time vector for simulating the ODE set, $\boldsymbol{\theta}$ is a p -dimensional vector of unknown parameters that will be estimated and \mathbf{x}_0 is known initial conditions for the model.

Model parameters $\boldsymbol{\theta}$ are estimated by iterative algorithms that minimises an *objective function*: the sum square error between experimental data $\mathbf{x}(t)$ and simulated data from model $\hat{\mathbf{f}}(\mathbf{x}(t), \boldsymbol{\theta})$ as shown in equation 3, where $i \in \{1, \dots, n\}$ represents state variable numbers, $l \in \{1, \dots, L\}$ is the number of synthetic data for each state variables \mathbf{x}_i .

$$\min_{\boldsymbol{\theta}} \sum_{i=1}^{i=n} \sum_{l=1}^{l=L} [\mathbf{x}_i - \hat{\mathbf{f}}_i(\mathbf{x}(t), \boldsymbol{\theta})]^T [\mathbf{x}_i - \hat{\mathbf{f}}_i(\mathbf{x}(t), \boldsymbol{\theta})] \quad (3)$$

Optimisation based parameter estimation algorithms are aim at solving a least squares data fitting problem. Such concept can be illustrated using fungal model as shown

figure 5, where the error between experimental data \mathbf{x} and model simulation with estimated parameters $\hat{\mathbf{f}}(\mathbf{x}(t), \hat{\boldsymbol{\theta}})$ as represented in red bars is minimum. This is equivalent to the formulation of the objective function (equation 3), since both of the graphical illustration and the objective function are intending to return the parameter values which resultant the smallest sum of squared error (SSE) between data and model simulation.

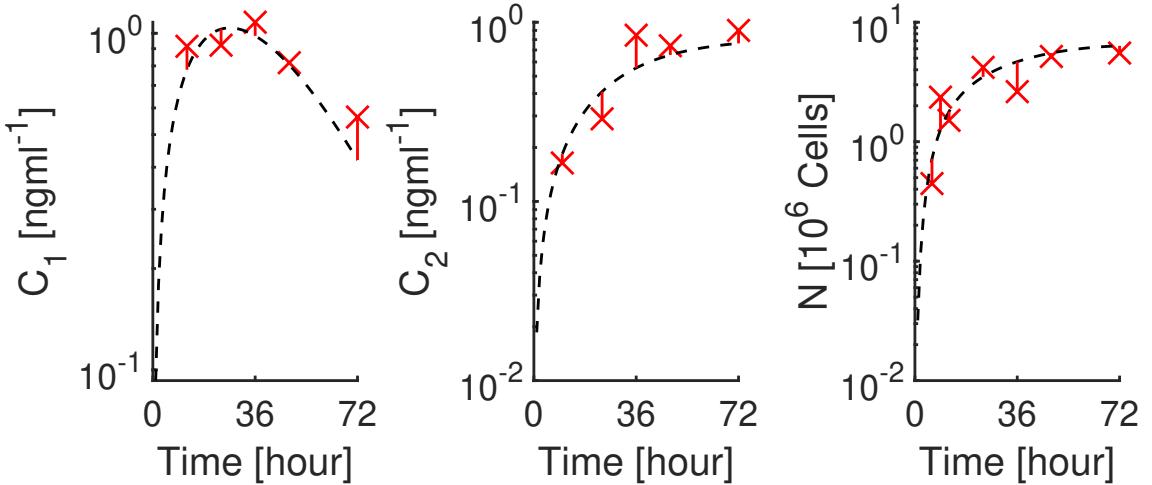


Figure 5: Illustration of parameter estimation using mathematical optimisation. The error (—) between fungal model simulation (---) and experimental data (×) is minimum with output parameters from the optimisation algorithms.

Both genetic algorithm and Gauss-Newton iteration are iterative optimisation based parameter estimation algorithms. Their individual formulation, modification accounting for the features of synthetic data and selection of hyper-parameters will be introduced and justified in the remaining subsections respectively.

3.2.1 Genetic Algorithm

Genetic algorithm initialises by randomly sampling N sets of parameters $\boldsymbol{\theta}$ with length p which is referred as *population*. The population of parameters evolves iteratively in comparison with natural selection process: the stronger parameters in the population survive where the strength of each set of parameters $\boldsymbol{\theta}$ is assessed by the objective function (equation 3). The survived population are referred as *parents* who have opportunities in producing new parameters named *children*. These survived parents and children update the population pool in the next iteration (referred as *generation*), and their survivability are assessed in the same way. Over generations, the parameters which have a smaller objective function value are expected to be retained or to evolve.

The process of updating generations in order to choose optimal parameters $\boldsymbol{\theta}$ is illustrated in figure 6, where k is generation number and square brackets indicate the dimension of a matrix.

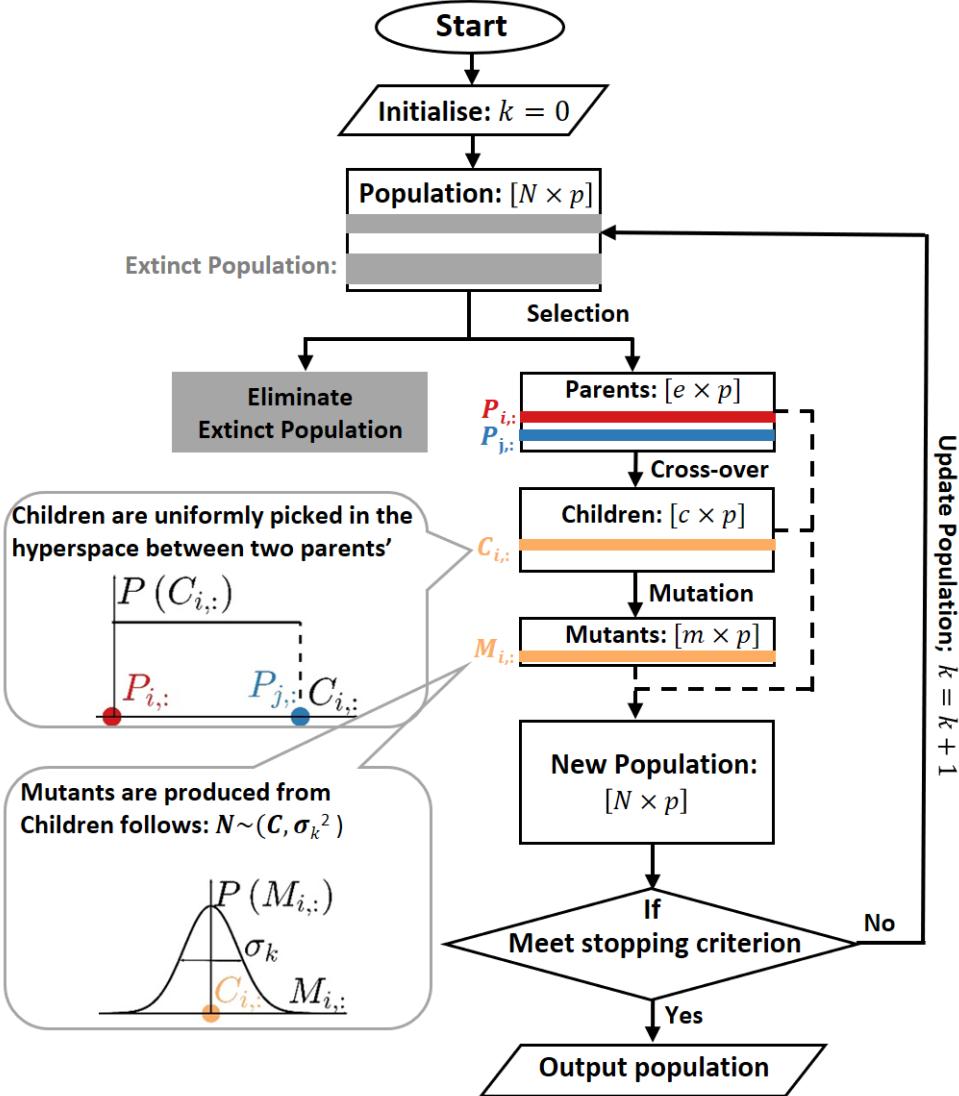


Figure 6: Illustration of parameter estimation using genetic algorithm through updating population iterative . In each generation, the population is updated by selection, cross-over and mutation operations until stopping criterion is met, where selection is a process to separate extinct populations and parents. Then the hyperspace between parents will be explored through cross-over and mutation operations.

As shown in figure 6, the population $[N \times p]$ is formed by N set of parameters of length p . By ranking the population's objective function values in ascending order, a set of parameter will be either selected as a parents ($P_{i,:}$, $i \in \{1, \dots, e\}$, where ' $i,:$ ' represents the i^{th} row of a matrix) or becoming extinct. The parents will have opportunities in producing new parameters (children) through *cross-over* operation as:

$$C_{i,:} = P_{i,:} + \mathbf{U} \sim (0, 1) \times [P_{j,:} - P_{i,:}] \quad (4)$$

where C is children and $i \in \{1, \dots, c\}$. The cross-over operator is responsible for producing hopefully better performing children who can fit the model better to the data by exploring the hyperspace between parents.

The children are allowed to move around following a zero mean normal distribution, such operation is referred as *mutation* and is defined as shown in equation 5, where M is mutated children namely *mutants*, G indicates the maximum generation number. The stochastic mutation operation can be crucial for the algorithm being able to escape from a local minimum, since the operation is independent from objective function values.

$$M_{i,:} = C_{i,:} + \mathcal{N} \sim (0, \sigma_k), \text{ where :} \\ \sigma_k = \sigma_{k-1} \left(1 - \frac{k}{G}\right) \quad (5)$$

The stopping criteria determine whether to update populations for the next generation or to terminate the algorithm and output the population from the current generation. The algorithm is expected to terminate when any of the condition is met: (i) maximum generation number (G) is 200, since there is no significant improvement in average objective function values even after generation number 20 in trial run (figure 7), (ii) the relative change in standard deviation of the population objective function fail below a tolerance level, which suggests that there is no further improvement possible.

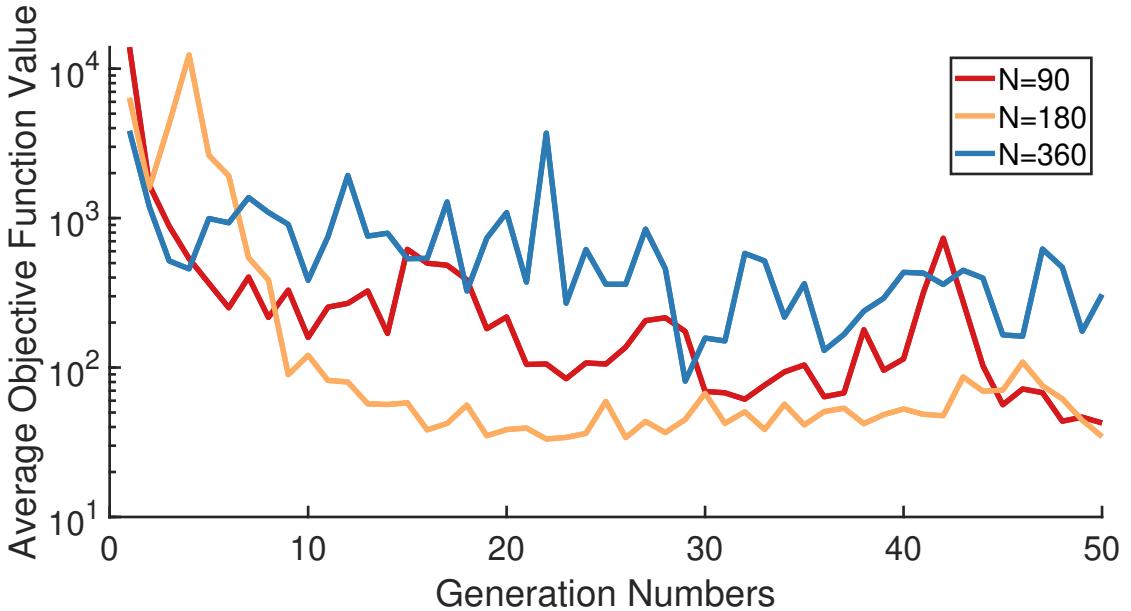


Figure 7: trial parameter estimation of fungal model using MATLAB $ga()$ function with population size $N = 90, 180$ and 360 , respectively. Their run time are $19, 41$ and 84 minutes, respectively. Stopping criterion (i) maximum generation number (k) is set as 200 , since there was no significant improvement in average objective function values in all $N = 90, 180$ and 360 . even after the 20^{th} generation

The population number N was selected as 180 in simulation study by taking into account the efficiency in reducing average objective function values as shown in figure 7 and the computation time. When $N = 180$, the average objective function values after 10 generations fluctuate the least. On the contrary, $N = 360$ results exponentially growing computation, yet the average objective function value trace were even not improved comparing to that of $N = 90$ and 180 .

During our simulation study using genetic algorithm, the intrinsic objective function values at each generation were also compared an reference value, which were computed from a mean value model defined as:

$$\sum_{i=1}^{i=n} \sum_{l=1}^{l=L} (\mathbf{x}_i(t) - \bar{\mathbf{x}}_i)^2 \quad (6)$$

where $i \in \{1, \dots, n\}$ represents state variable numbers, $l \in \{1, \dots, L\}$ is the number of synthetic data for each state variables \mathbf{x}_i , $\bar{\mathbf{x}}_i$ is the mean value of the synthetic data for each state variables. The reference model can be visualised as seen in figure 8.

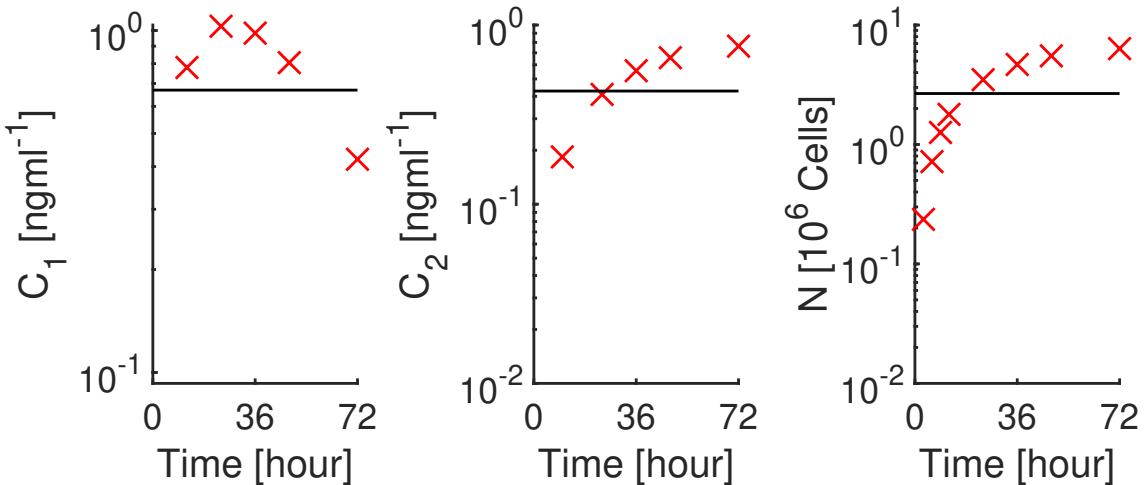


Figure 8: Reference model (—) defined in equation 6 for computing an reference value (47.15) from synthetic data (×). The reference value acts as a baseline mark in comparison with fungal model objective function values over generations in the simulation study

Genetic algorithm is one of the bioinspired computational intelligence commonly used for parameter estimation [9]. It is superior when the objective function gradient is intractable [29], since selection, cross-over and mutation are three gradient-free operations drive the algorithm in exploring the parameter space.

3.2.2 Gauss-Newton Iteration

Gauss-Newton algorithm is one of the classical gradient-based optimiser in solving least squares minimisation problems [29]. It is expected to have fast convergence property when the algorithm is initialised at the vicinity of the global minimum. Different with genetic algorithm, gradient of the objective function determines the travelling of the algorithm in parameter space.

At each algorithm iteration k , the algorithm revises parameter values for the upcoming step as shown in figure 9, where the algorithm is initialised with one set of parameter $\theta_0 \in \mathbb{R}^p$.

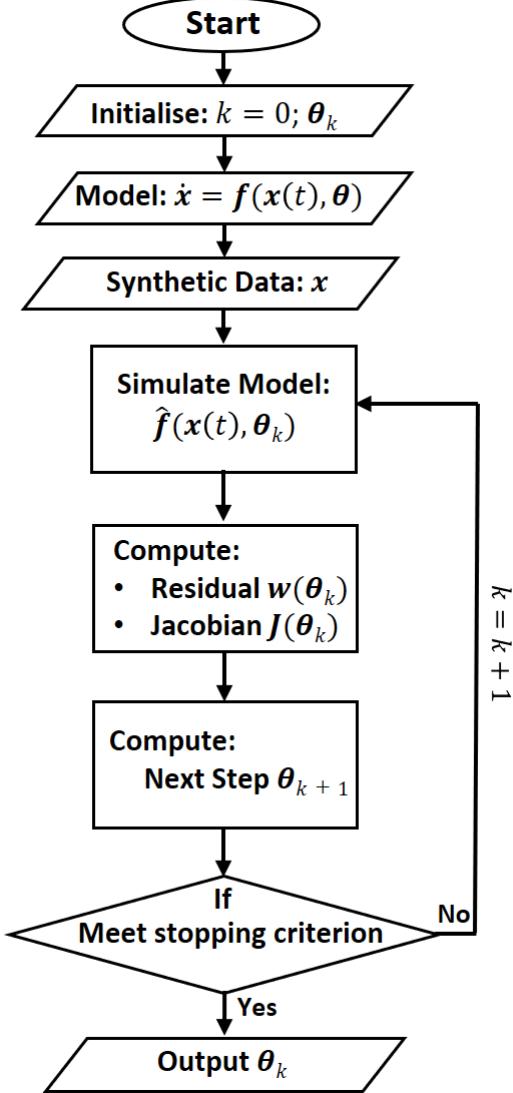


Figure 9: Flow chart of iterative Gauss-Newton iteration that estimates parameters θ for ODE model $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), \theta)$. Residual function (equation 7) and Jacobian function (equation 8) will be computed from model simulation $\mathbf{f}(\mathbf{x}(t), \theta)$ and synthetic data \mathbf{x} . These information will be used to guide the exploration in parameter space for the upcoming iteration until stopping criterion is met.

The model is then numerically simulated with parameter values in current iteration θ_k . This will allow us to compute residual function \mathbf{w} as:

$$w_l(\theta^k) = \frac{1}{2} \sum_{i=1}^{i=n} [\mathbf{x}_i(t = t^*) - \hat{\mathbf{f}}_i(\mathbf{x}(t = t^*), \theta^k)]^2 \quad (7)$$

where $\mathbf{w} \in \mathbb{R}^L$, $l \in \{1, \dots, L\}$ and L is the number of synthetic data for each state variable. However, L is different for each state variable in our simulation study, since the synthetic data was designed to maintain non-simultaneously measured state variables feature from previously encountered experimental data as discussed in section 3.1. As a consequence, the residual function cannot be computed at this stage since equation 7 requires synthetic data for all state variables $i \in \{1, \dots, n\}$ at a given time

$t = t^*$. In order to evaluate residual function, the solution here is to assume the latent synthetic data at a given time $\mathbf{x}_i(t^*)$ equals to the simulated data from the model $\hat{\mathbf{f}}(\mathbf{x}(t^*), \boldsymbol{\theta}^k)$. In our simulation, the discretized time $t^* = [0, 3, 6, 9, 10, 12, 24, 36, 28, 72]$ hour and $L = 10$, since synthetic data from at least one state variable exists at these time points as shown in figure 4.

The gradient of the objective function is generate by Jacobian as:

$$\mathbf{J}_{:,j} = \left. \frac{\partial \left[\sum_{i=1}^{i=n} f_i((\mathbf{x}(t = t^*), \boldsymbol{\theta})) \right]}{\partial \theta_j} \right|_{\boldsymbol{\theta}=\boldsymbol{\theta}^k} \quad (8)$$

where $\mathbf{J} \in \mathbb{R}^{L \times p}$, p is the number of parameters, ‘ $:,j$ ’ indicates the j^{th} column of a matrix, $j \in \{1, \dots, p\}$, $f_i \in \mathbb{R}^L$, $i \in \{1, \dots, n\}$ and n is the number of state variables.

Gauss-Newton iteration exploits the residual function and Jacobian as a searching direction and compute the upcoming step in parameter space as:

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{w}(\boldsymbol{\theta}^k) \quad (9)$$

The algorithm is expected to iteratively update the parameter values in the upcoming step until meeting the stopping criterion. In our study, the algorithm will terminate if any of the following metric is met: (1) maximum iteration number and (2) maximum iteration time are set as 100 and 100 seconds, respectively in order to avoid heavy computation. (3) maximum residual function magnitude is set as 1×10^{-2} since $\boldsymbol{\theta}$ is expected to converge at local minimum when $\|\mathbf{w}\|_2^2$ is close to zero (equation 9).

3.3 ODE Solving

As discussed in section 3.2, both of genetic algorithm and Gauss-Newton iteration requires computations in solving ODE recursively. MATLAB ODE solver *ode45()* is chosen by assuming the ODE model is non-stiff. Positive results are enforced since the testing model (equation 1) represents the concentration of different species.

In addition to the selection of the ODE solver, the maximum solving time for every ODE simulation is capped at 1 second. This action prevents the algorithm being trapped in *unsolvable region*, that is when the ODE is computationally heavy to integrate with some parameter values. The maximum solving time 1 second is sufficient, since the it took 7.29×10^{-4} seconds on average and with range of $[4.92 \times 10^{-4}, 1.98 \times 10^{-1}]$ seconds to solve fungal model using nominal parameter values displayed in table 1 over 10000 runs (figure 10a).

As a consequence of capping the maximum solving time for the ODE solver, model simulation longer than the limited time will lead to a non-numerical objective function value (equation 3) due to the inaccessible $\hat{\mathbf{f}}_i(\mathbf{x}_i(t), \boldsymbol{\theta})$. This is seen as the ‘Nan’ category in figure 10b, which is a histogram plot of the fungal model objective function

values with 10000 set of independently sampled parameters from $\mathbf{U} \sim [0, 1]$ solved by capped *ode45()*.

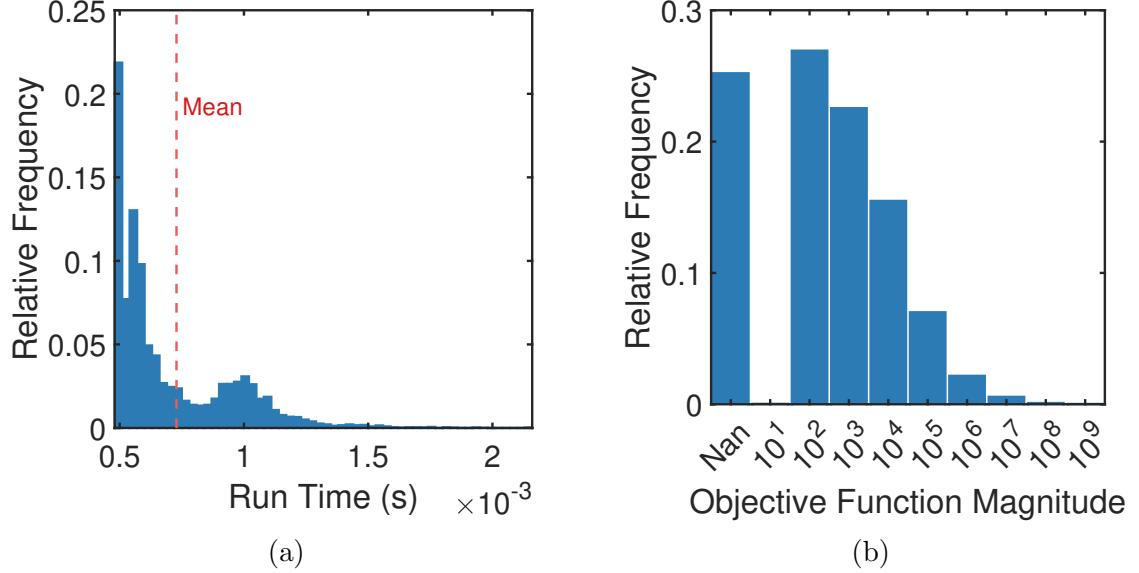


Figure 10: (a): *ode45()* run time distribution for solving fungal model (equation 1) using nominal parameter values (table 1) over 10000 runs with mean (---) of 7.29×10^{-4} seconds and with range of $[4.92 \times 10^{-4}, 1.98 \times 10^{-1}]$ seconds. The distribution displayed [0, 99.9] percentile of data for the visualisation purpose. (b): The objective function distribution of 10000 set of independently sampled parameter values from $\mathbf{U} \sim [0, 1]$ which are categorised based on magnitude. The ‘Nan’ category are the runs that sampled parameters were unsolvable to the ODE solver within 1 second. Over 90% of these objective function values are below the magnitude of 10^5 .

The objective function value is prevalent for genetic algorithm selection process, which confines the parameter searching space as discussed in section 3.2.1. These non-numerical objective function values due to unfavorable parameters were set as 10^5 for two purposes: (i) the value is large enough so the unsolvable parameters from a population go extinct during the selection process, (ii) the value is reasonable and do not cause discontinuity in objective function surface, hence avoiding interference with stopping criteria. The value 10^5 was chosen since 90% of objective function values are below the magnitude of 10^5 in figure 10b. On the other hand, these non-numerical objective function values were not manipulated with Gauss-Newton iteration. This is because of the searching direction at each iteration $(\mathbf{J}^T \mathbf{J})^{-1} \mathbf{J}^T \mathbf{w}(\boldsymbol{\theta}^k)$ in equation 9 is mathematically derived descending direction.

4 Results

4.1 Genetic Algorithm

The initial population composed of 180 set of parameters with length 9 (as discussed in section 3.2.1) and each of the initial population element was sampled independently from $\mathbf{U} \sim [0, 1]$. Genetic algorithm was terminated due to maximum generation number 200 was reached. The results that support our finding in terms of (i) nominal parameter values retrieval, (ii) convergence to local minimum, (iii) curve fitting and (iv) parameter non-identifiability are displayed, respectively.

4.1.1 Parameter Retrieval

The estimated 180 parameter sets at the 200th generation are displayed in the correlation matrix plot (figure 11): the diagonal plots are parameter value distributions. The lower triangular area shows estimated parameters' bivariate distribution (•) and the nominal parameter values (×). The upper triangular area shows the correlation coefficients (R) between pairs of parameters.

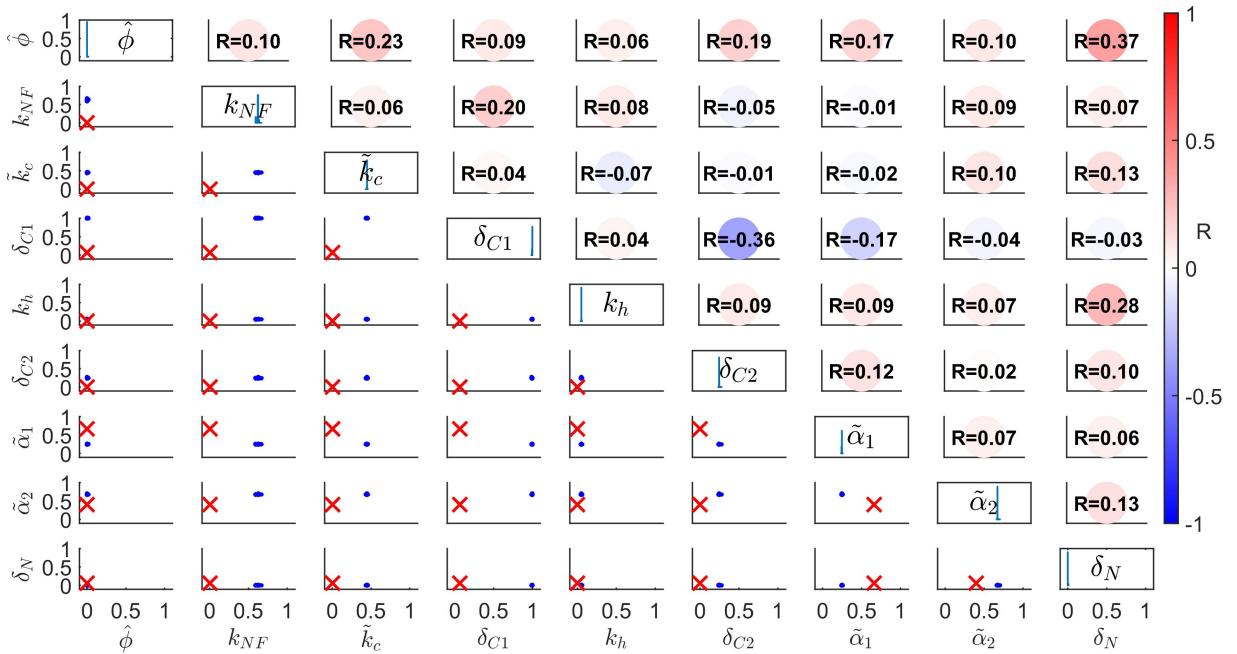


Figure 11: Population at genetic algorithm termination which consists of 180 set of parameters. The diagonal plots are parameter distributions with corresponding parameters labelled. The lower triangular plots are pairwise scatter plots (•) and the pairwise nominal parameter values (×). The parameter pairs can be read from the position of the subplots, where the column and the row indicate the parameter being plotted on x and y axis, respectively. The upper triangle panels are the correlation coefficients (R) corresponding to the parameter pairs of the scatter plots at the opposite side along the diagonal. The intensity of the coloured circles in upper triangular area is proportional to the magnitude of R.

The result displayed in figure 11 lower triangular plots suggests that genetic algorithm

was not able to recover nominal parameter values using sparse synthetic data, since the distribution of the estimated parameters does not match with nominal parameter values.

4.1.2 Local Minimum

The objective function values history (equation 3 at each generation) is displayed in figure 12, where both average and minimum objective function values have not been significantly reduced since the 20th generation given linearly increased computational time. This suggests that the objective function values are unlikely to be further reduced even the algorithm was not terminated at generation number 200.

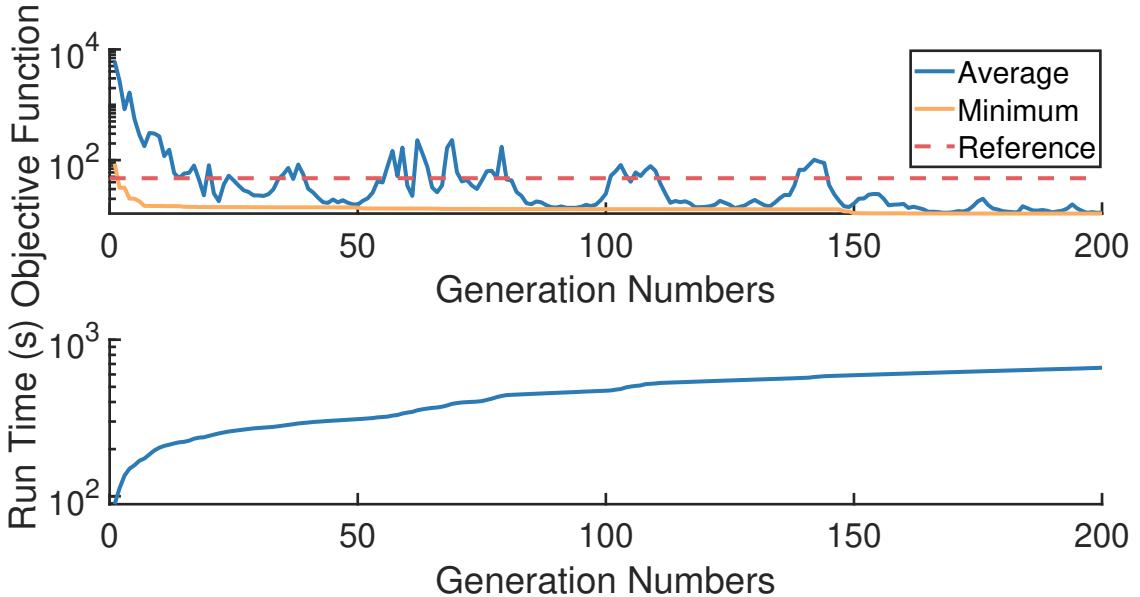


Figure 12: The objective function values and computational time at each generation for the simulation study using genetic algorithm. The reference value (equation 6 and figure 8) acts as a baseline mark.

The population evolution across generations have also been illustrated by displaying population for parameters $(\hat{\phi}, k_{NF}, \tilde{k}_c)$ at generation number 1, 15, 30 and 200 in figure 13 (The full matrix plots can be found in Appendix B for generation number 1, 15 and 30, and figure 11 for generation number 200). The selection of displayed generation number is based on objective function history as shown in figure 12, where the average objective function values reduced significantly in the first 50 generations due to significantly changed populations. The lower triangular area of each subplot in figure 13 shows that parameters start to cluster at an area other than its nominal parameter values. Moreover, most of parameters show uni-modal distribution and weak pairwise correlations ($|R| < 0.3$) apart from the following parameter pairs: $[\delta_{C1}, \delta_{C2}]$ and $[\hat{\phi}, \delta_N]$ at generation 200 (as seen in figure 11 upper triangular area). These indicate that the genetic algorithm is trapped at a local minimum at generation 200.

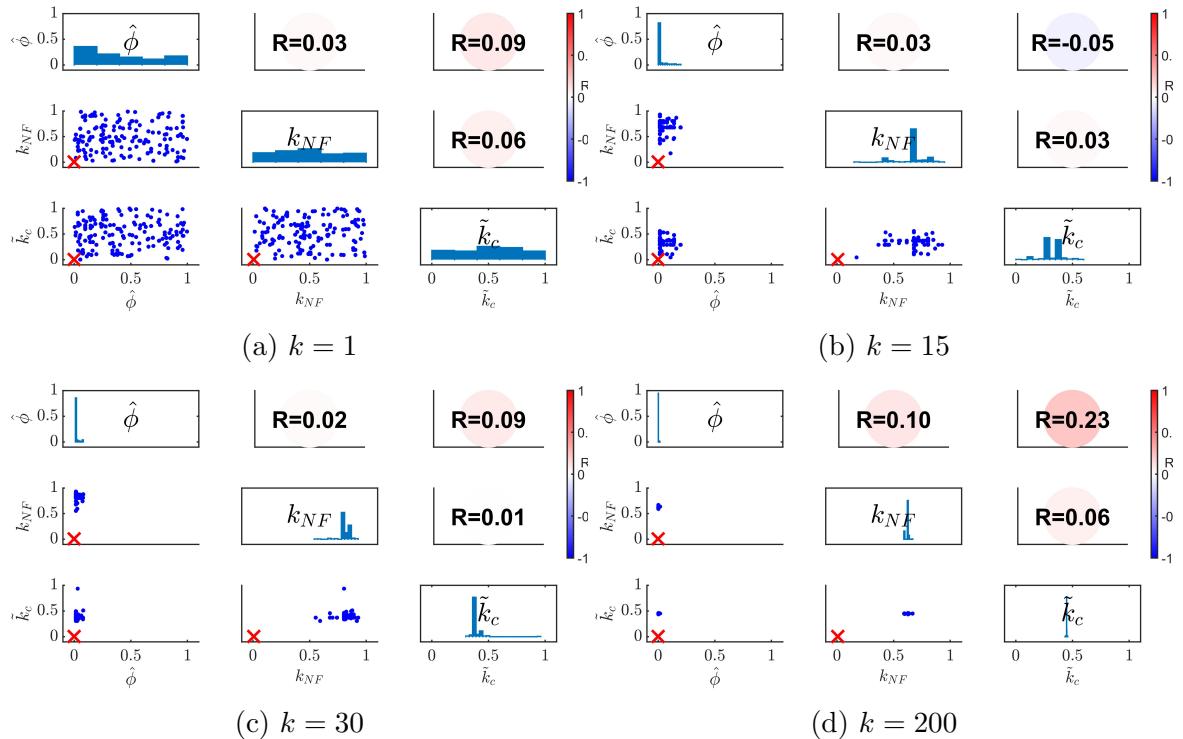


Figure 13: Genetic algorithm population (•) at generation number $k = 1, 15, 30$ and 200 for selected three parameters. The complete distribution for $k = 1, 15$, and 30 can be found in Appendix B and for $k = 200$ can be found in figure 11. The lower triangular area shows the population starts to cluster in a region other than their nominal parameter values (✗), which suggests the algorithm converges to a local minimum.

4.1.3 Curve Fitting

Fungal model simulation using *best performing parameters*, which are the parameters resulting in the minimum objective function values at each generation, are displayed in figure 14, where the intensity of the model simulation trace is proportional to generation numbers. It is perceived that the genetic algorithm attempted to fit the fungal model to state variable N but ignored C_1 and C_2 .

The sum squared error (SSE) between synthetic data and model simulation for each state variables are also displayed in figure 15, where SSE for N is significantly reduced although it is still marginally higher than the SSE of both C_1 and C_2 .

From both figure 14 and figure 15: It is convincing that a greater magnitude of objective function (equation 3, i.e. sum of all state variables' SSE) was derived due to failure in fitting model trajectory to state variable N at initial generations (less intense trace in figure 14 and SSE at initial generations in figure 15). While neglecting curve fitting to state variables C_1 and C_2 have negligible effect on objective function values as seen at the later generations. These suggest that SSE might not be an appropriate objective function for fungal model.

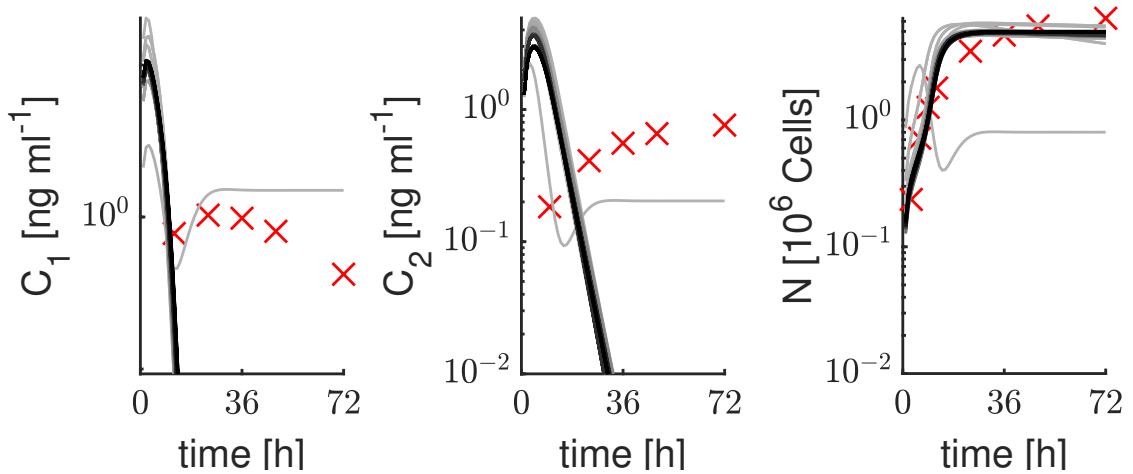


Figure 14: Synthetic data (\times) and fungal model simulation using best performing parameters from each generation (—), where the intensity of the solid line is proportional to generation numbers.

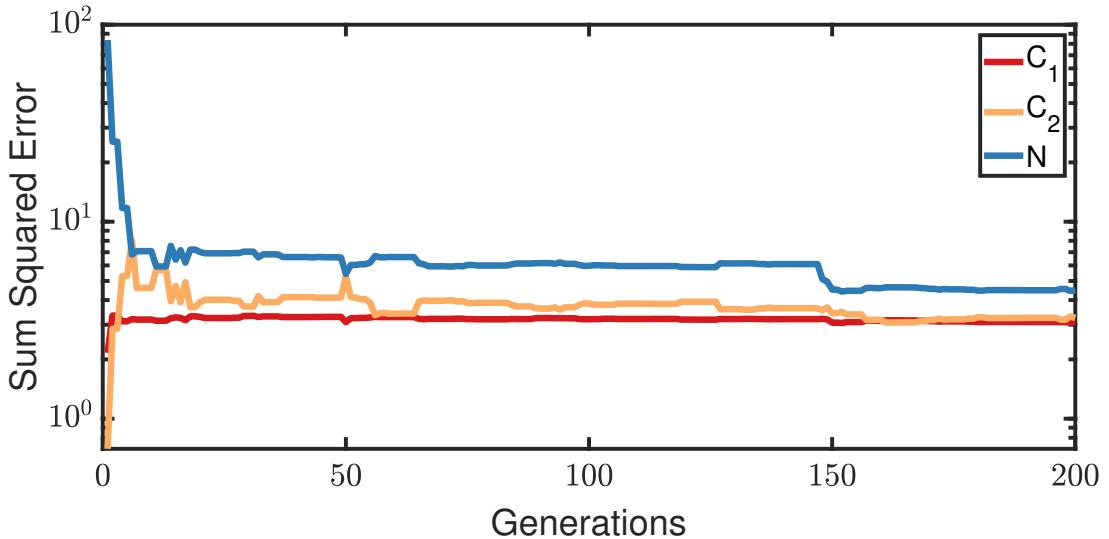


Figure 15: Sum squared error between synthetic data and model simulation using best performing parameters for state variables C_1 , C_2 and N , respectively.

4.1.4 Parameter Non-identifiability

Figure 16 shows the distribution of the 200 best performing parameters. It was found that there exists a high correlation ($|R| > 0.7$) between several best performing parameter pairs as shown in figure 16 upper triangular area. In addition, the pairwise distributions in the lower triangular area plots suggest that there are non-uniquely defined parameter sets resultant to considerably similar objective function values (ranging in between 10.8 and 14.9 after the 7th generation as shown in figure 12). Moreover, the model simulation using these non-uniquely defined best performing parameter shows identical features in their trajectories as shown in figure 17. These observations raise a question on the fungal model's identifiability, that is whether parameters can be uniquely defined under the current sparse synthetic data set up.

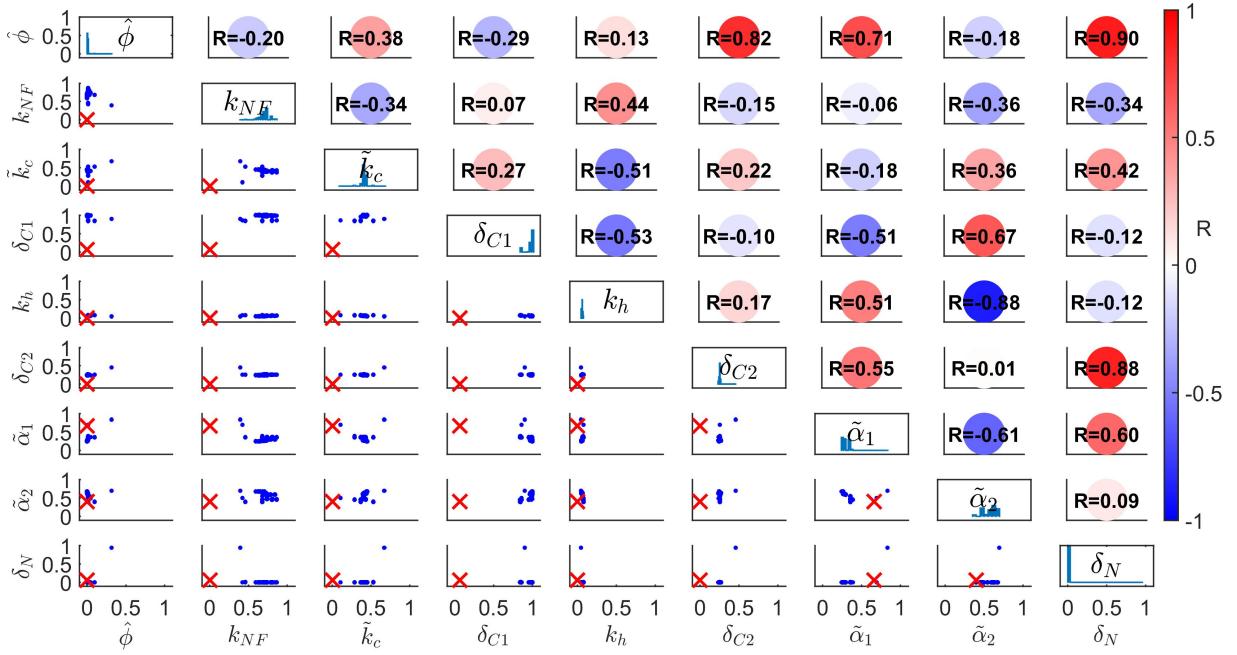


Figure 16: Distribution and correlations of 200 set of best performing parameters, which are the parameter resultant in the minimum objective functions at each generation. Given these parameters resultant a considerably similar objective function values (more than 95% of parameters have an objective function value in the range of [10.8, 14.9]), the non-unique parameter distribution and strong correlations suggests that the parameters cannot be uniquely defined.

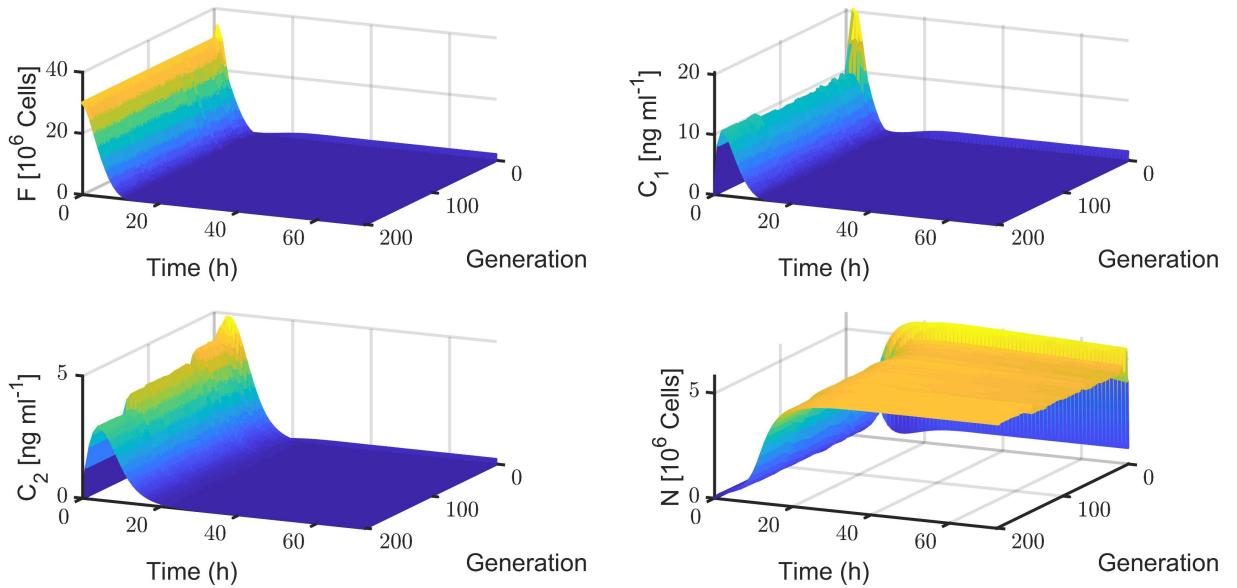


Figure 17: Fungal model simulation using the best performance parameters at each generation. Although these parameters are located at different geometry in parameter space, their simulation trace shows the same feature.

4.2 Gauss-Newton Iteration

The results from Gauss-Newton iteration were initialised in two different ways in order to demonstrate our findings in terms of (i) parameter retrieval and (ii) model instability, respectively.

4.2.1 Parameter Retrieval

The Gauss-Newton iteration was initialised by sampling a set of parameter (corresponding to θ_0 in figure 9 from section 3.2.2), and each single parameter are sampled independently from $\mathbf{U} \sim [0, 1]$. There were 10000 runs performed in parallel. All the runs were terminated before meeting any stopping criteria defined in section 3.2.2. This was attributed to that the fungal model falls into unsolvable parameter space and hence the evaluation of the next step in parameter space (equation 7-9) became unavailable. All 10000 runs terminated before iteration 4 as shown in the left panel of figure 18, where termination iteration (N) is the total number of iteration before the algorithm breaks.

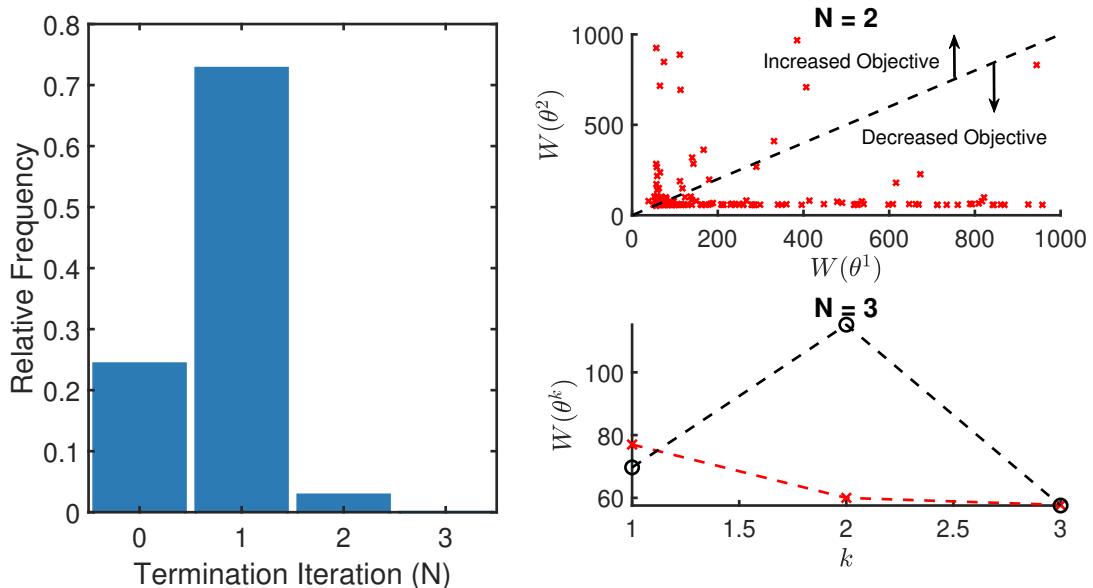


Figure 18: Left panel: relative frequency of Gauss-Newton iteration over 10000 runs with respect to termination iteration number (N), where parameter θ are initialised from $\mathbf{U} \sim [0, 1]$. All runs were terminated before iteration 4 due to the fungal model falls into unsolvable parameter space. There were 2/10000 runs terminated after iteration 3 which are invisible in the frequency plot. Right panel: the objective function histories for $N = 2$ and 3, where $W(\theta^k)$ is the objective function evaluated at parameter at the k^{th} iteration, there were 212/273 runs that W was reduced before termination.

The simulation study illustrated that it is unavoidable to encounter unsolvable parameters from both algorithm initialisation (figure 18 left panel $N=0$ category) and gradient directed parameter space exploration (figure 18 left panel $N=1, 2, 3$ categories). These unsolvable parameters causes inability in computing objective function gradient, hence the iteration is forced to break down. Therefore, it is reasonable to

conclude that fungal model parameters cannot be retrieved under sparse and irregular synthetic data set up by using Gauss-Newton iteration.

4.2.2 Model Instability

Figure 18 right panel shows the objective function histories for $N = 2$ and 3 . It suggests that the Gauss-Newton iteration does achieve its aim, that is to reduce objective function values iteratively, since $212/273$ runs experienced reduction in objective function values for $N = 2$ and 3 before termination. At this stage of understanding, we believe that the instability of the ODE model make gradient-based algorithm impracticable for fungal model parameter estimation.

Parameter estimation using Gauss-Newton iteration was also performed by initialising at the vicinity of the nominal parameter values to confirm the instability of the model. Parameters were initialised as:

$$\theta_0^i \sim \mathcal{N}(\theta_i, \sigma_i^2) \quad (10)$$

where θ_0^i is the i^{th} parameter at iteration $k = 0$, $i \in \{1, \dots, p\}$, p is the number of parameters, $\boldsymbol{\theta}$ is nominal parameter values and σ_i is standard deviation for each θ_i , which is defined as:

$$\sigma_i = 10^{\lfloor \log_{10} \theta_i \rfloor} \quad (11)$$

In this way, each parameter was initialised around its nominal parameter values by a perturbation relative to its own nominal parameter magnitude. 10000 runs were performed and all of them fail due to the ODE entered unsolvable region, which further verified the instability of the fungal model around nominal parameter values.

5 Discussion

The simulation study using both stochastic global optimiser genetic algorithm and gradient-based local optimisation algorithm Gauss-Newton iteration shows that either of those could not retrieve the nominal parameter values using sparse synthetic data. In addition, each of the algorithms was unsuccessful for their own reasons: the genetic algorithm converged to a local minimum and Gauss-Newton iteration broke down unexpectedly due to the algorithm entering the model’s unsolvable regions. The results displayed in section 4 have flagged signs of major factors that impede parameter estimation algorithm converging to nominal parameter values: *model identifiability* and *model stability*. In another words, the unsuccessful parameter retrieval in our simulation study was not solely attributed to data sparsity.

Model identifiability is a concept regarding whether all parameters can be uniquely defined from a given model and given experimental measurements [61]. Such property is predominated by the model structure and experimental data availability [62]. Our simulation studies using the genetic algorithm with sparse synthetic data have shown convergence to local minimum and the existence of multiple highly correlation parameter pairs, which suggests the probable existence of non-identifiable parameters. Indeed, the failure in convergence using multiple parameter estimation algorithms could also be a prognosis of system non-identifiability [63]. Several diagnostic algorithms have been developed which could subsequently alter the ill-posed problem into solvable. These frameworks include parameter reduction by grouping [63–65] or fixing the value of non-identifiable parameters [2, 66, 67] for problems caused by model structure. Another approach is optimal experiment design for tackling non-identifiability caused by the lack of experimental data [68, 69].

Model stability is commonly recognised as system trajectory sensitivity to a small perturbation around nominal parameter values. The simulation study using Gauss-Newton iteration shows that frequent issues arose due to the algorithm falling into ODE unsolvable regions, which calls into question the stability of the fungal model. This was further confirmed by that the persistently encountered unsolvable regions even when initialising the Gauss-Newton iteration at the vicinity of the nominal parameter values. Such study also demonstrated that parameter estimation for a highly unstable ODE model using gradient-based iterative algorithm is unfavorable. Therefore, bifurcation analysis or simulation studies, which is relatively less computational laborious, is recommended prior to parameter estimation. Since it can be beneficial for selecting appropriate parameter estimation paradigms.

Additionally, the limitations of this study include: (i) we are unable to distinguish whether the non-identifiability for fungal model parameter estimation is attributed to model structure or due to the data sparsity, (ii) we cannot exclude the effect on unsuccessful parameter retrieval from the modification of the algorithms in order to handle non-simultaneous data at a given time for each state variables (section 3.2.2) and for avoiding ODE non-solvable region (section 3.3) at the algorithm implementation stage, (iii) this study did not consider data noise levels which are encountered in model calibration in general. Hence, further analysis needed for investigating the effect of these

factors on sparse data model calibration.

Finally, parameter estimation for well-posed problems can be more efficiently solved by hybrid optimisation strategy [4]: that is employing a stochastic global optimiser to confine the parameter space where global minimum is possibly contained, then switch to a gradient-based local minimiser in order to achieve fast convergence to the global minimum. Various hybrid optimisation algorithms have been developed and demonstrated for their efficiency in achieving global minimum convergence comparing to stochastic global optimiser and the ability of escaping from local minimum that a single gradient-based local minimiser cannot reach [70–74]. Such strategy can be potential solution for fungal model parameter estimation once model identifiability and stability issues are analysed and resolved.

6 Conclusion

In this project, a simulation study was conducted in order to investigate the attainability of sparse data model calibration using classical parameter estimation techniques. It was found that parameter retrieval was not possible using an invasive aspergillosis model with sparse and latent synthetic data. The simulation study suggests that the lack of data is not the only dominant factor that complicated the parameter estimation problem. Factors such as model structure and model stability, which are independent from data availability, should also be examined before thorough temptation in parameter estimation. Failure in considering model structure and model stability for general model calibration problem will result in unsuccessful parameter estimation using different algorithms and lead to unnecessary computational expenses. In terms of sparse data model calibration, the questions of whether currently available data for model calibration is sufficient can be diagnosed once the aforementioned two factors are precluded.

References

- [1] P. Mendes and D. Kell. Non-linear optimization of biochemical pathways: Applications to metabolic engineering and parameter estimation. *Bioinformatics (Oxford, England)*, 14:869–883, 1998. doi: 10.1093/bioinformatics/14.10.869.
- [2] R. Eftimie, J.J Gillard, and D.A. Cantrell. Mathematical Models for Immunology: Current State of the Art and Future Research Directions. *Bulletin of Mathematical Biology*, 78(10):2091–2134, 2016. doi: 10.1007/s11538-016-0214-9.
- [3] J. Banga. Optimization in computational systems biology. *BMC systems biology*, 2:47, 2008. doi: 10.1186/1752-0509-2-47.
- [4] A.R. Muhammad, S.Mohamad. Mohd, D. Safaai, N. Suhaimi, S. Richard, and F.S. Muhammad. Metaheuristic optimization for parameter estimation in kinetic models of biological systems - recent development and future direction. *Current Bioinformatics*, 12(4):286–295, 2017. doi: 10.2174/1574893611666161018142809.
- [5] K. Schittkowski. *Numerical Data Fitting in Dynamical Systems: A Practical Introduction with Applications and Software*. Kluwer Academic Publishers, USA, 2002. ISBN 9781441957627.
- [6] J.H. Holland. *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence*. Complex adaptive systems. MIT Press, Cambridge, USA, 1992. ISBN 9780262275552.
- [7] L.J Park, C.H Park, C. Park, and T. Lee. Application of genetic algorithms to parameter estimation of bioprocesses. *Medical and Biological Engineering and Computing*, 35(1):47–49, 1997. doi: 10.1007/BF02510391.
- [8] W. Banzhaf, F.D. Francone, R.E. Keller, and P. Nordin. *Genetic Programming: An Introduction: On the Automatic Evolution of Computer Programs and Its Applications*. Morgan Kaufmann Publishers Inc., San Francisco, USA, 1998. ISBN 155860510X.
- [9] M.S. Nobile, A. Tangherloni, L. Rundo, S. Spolaor, D. Besozzi, G. Mauri, and P. Cazzaniga. Computational Intelligence for Parameter Estimation of Biochemical Systems. In *2018 IEEE Congress on Evolutionary Computation, CEC 2018 - Proceedings*, pages 1–8, 2018. ISBN 9781509060177. doi: 10.1109/CEC.2018.8477873.
- [10] C. Modchang, W. Triampo, and Y. Lenbury. Mathematical modeling and application of genetic algorithm to parameter estimation in signal transduction: Trafficking and promiscuous coupling of G-protein coupled receptors. *Computers in Biology and Medicine*, 38(5):574–582, 2008. doi: 10.1016/j.combiomed.2008.02.005.
- [11] T. Tian. Estimation of kinetic rates of MAP kinase activation from experimental data. In *Proceedings - 2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing, IJCBs 2009*, pages 457–462, 2009. doi: 10.1109/IJCBs.2009.78.

- [12] A. Pankajakshan, S.M. Pudi, and P. Biswas. Acetylation of Glycerol over Highly Stable and Active Sulfated Alumina Catalyst: Reaction Mechanism, Kinetic Modeling and Estimation of Kinetic Parameters. *International Journal of Chemical Kinetics*, 50(2):98–111, 2018. doi: 10.1002/kin.21144.
- [13] A. Naeimifard and A. Ghaffari. Computational model of recirculating lymphocytes. *Journal of Biological Systems*, 23(1):93–114, 2015. doi: 10.1142/S0218339015500060.
- [14] C. Wang and L. Zhang. Comparison of parameters estimation methods based on the systems biology model of breast cancer. In *Proceedings of 2015 IEEE International Conference on Progress in Informatics and Computing, PIC 2015*, pages 561–565, 2016. doi: 10.1109/PIC.2015.7489910.
- [15] R. Singh, S.S. Miriyala, L. Giri, K. Mitra, and V.V. Kareenahalli. Identification of unstructured model for subtilin production through *Bacillus subtilis* using hybrid genetic algorithm. *Process Biochemistry*, 60:1–12, 2017. doi: 10.1016/j.procbio.2017.06.005.
- [16] C. Sepúlveda, O. Montiel, J.M. Cornejo, and R. Sepúlveda. Estimation of population pharmacokinetic model parameters using a genetic algorithm. In *Advances in Intelligent Systems and Computing*, volume 648, pages 214–221. Springer Verlag, 2018. doi: 10.1007/978-3-319-67137-6_23.
- [17] A. Palit, S.K. Bhudia, T.N. Arvanitis, G.A. Turley, and M.A. Williams. In vivo estimation of passive biomechanical properties of human myocardium. *Medical and Biological Engineering and Computing*, 56(9):1615–1631, 2018. doi: 10.1007/s11517-017-1768-x.
- [18] A. Fatemi and H. Mahmoodian. Error dynamic shaping in HIV optimized drug delivery control. *Evolving Systems*, pages 1–14, 2020. doi: 10.1007/s12530-020-09329-2.
- [19] S.J. Wu and C.T. Wu. Computational optimization for S-type biological systems: Cockroach genetic algorithm. *Mathematical Biosciences*, 245(2):299–313, 2013. doi: 10.1016/j.mbs.2013.07.019.
- [20] M. Stražar, M. Mraz, N. Zimic, and M. Moškon. An adaptive genetic algorithm for parameter estimation of biological oscillator models to achieve target quantitative system response. *Natural Computing*, 13(1):119–127, 2014. doi: 10.1007/s11047-013-9383-8.
- [21] M. Ghovvati, G. Khayati, H. Attar, and A. Vaziri. Kinetic parameters estimation of protease production using penalty function method with hybrid genetic algorithm and particle swarm optimization. *Biotechnology and Biotechnological Equipment*, 30(2):404–410, 2016. doi: 10.1080/13102818.2015.1134279.
- [22] J.E. Lara-Ramirez, C.H. Garcia-Capulin, M.d.J Estudillo-Ayala, J.G. Avina-Cervantes, R.E. Sanchez-Yanez, and H. Rostro-Gonzalez. Parallel hierarchical genetic algorithm for scattered data fitting through B-splines. *Applied Sciences (Switzerland)*, 9(11):2336, 2019. doi: 10.3390/app9112336.

- [23] N. Kalogerakis and R. Luus. Improvement of gauss-newton method for parameter estimation through the use of information index. *Industrial & Engineering Chemistry Fundamentals*, 22(4):436–445, 1983. doi: <https://doi.org/10.1021/i100012a015>.
- [24] T. Lohmann, H.G. Bock, and J.P. Schloeder. Numerical methods for parameter estimation and optimal experiment design in chemical reaction systems. *Industrial & Engineering Chemistry Research*, 31(1):54–57, 1992. doi: <http://dx.doi.org/10.1021/ie00001a008>.
- [25] W. Tang, L. Zhang, A.A. Linniger, R.S. Tranter, and K. Brezinsky. Solving kinetic inversion problems via a physically bounded Gauss - Newton (PGN) method. *Industrial & Engineering Chemistry Research*, 44(10):3626–3637, 2005. doi: <http://dx.doi.org/10.1021/ie048872n>.
- [26] J.O. Ramsay, G. Hooker, D. Campbell, and J. Cao. Parameter estimation for differential equations: A generalized smoothing approach. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 69(5):741–796, 2007. doi: <10.1111/j.1467-9868.2007.00610.x>.
- [27] J. Bonilla, M. Diehl, F. Logist, B. De Moor, and J. Van Impe. An automatic initialization procedure in parameter estimation problems with parameter-affine dynamic models. *Computers and Chemical Engineering*, 34(6):953–964, 2010. doi: <10.1016/j.compchemeng.2009.10.020>.
- [28] G. Queisser and G. Wittum. A method to investigate the diffusion properties of nuclear calcium. *Biological Cybernetics*, 105(3–4):211–216, 2011. doi: <10.1007/s00422-011-0452-8>.
- [29] C. Zhao, Q. Xu, S. Lin, and X. Li. Hybrid differential evolution for estimation of kinetic parameters for biochemical systems. *Chinese Journal of Chemical Engineering*, 21(2):155–162, 2013. doi: [10.1016/S1004-9541\(13\)60453-X](10.1016/S1004-9541(13)60453-X).
- [30] D.P Moualeu, S. Bowong, and J. Kurths. Parameter Estimation of a Tuberculosis Model in a Patchy Environment: Case of Cameroon. In *Biomat 2013 - International Symposium On Mathematical And Computational Biology*, pages 352–373. World Scientific Pub Co Pte Lt, 2014. doi: 10.1142/9789814602228_0021.
- [31] I. Chervoneva, B. Freydin, B. Hipszer, T.V. Apanasovich, and J.I. Joseph. Estimation of nonlinear differential equation model for glucose-insulin dynamics in type i diabetic patients using generalized smoothing. *Annals of Applied Statistics*, 8(2):886–904, 2014. PT: J; UT: WOS:000342407200010.
- [32] E.J. Mansell, P.D. Docherty, L.M. Fisk, and J. Geoffrey Chase. Estimation of secondary effect parameters in glycaemic dynamics using accumulating data from a virtual type 1 diabetic patient. *Mathematical Biosciences*, 266:108–117, 2015. doi: <10.1016/j.mbs.2015.06.002>.
- [33] J.C. Ryan, K.W. Dunn, and B.S. Decker. Effects of chronic kidney disease on liver transport: Quantitative intravital microscopy of fluorescein transport in the

- rat liver. *American journal of physiology. Regulatory, integrative and comparative physiology*, 307(12):R1488–R1492, 2014. doi: 10.1152/ajpregu.00371.2014.
- [34] M. Tavakoli, K. Tsekouras, R. Day, K.W. Dunn, and S. Pressé. Quantitative Kinetic Models from Intravital Microscopy: A Case Study Using Hepatic Transport. *The Journal of Physical Chemistry B*, 123(34):7302–7312, 2019. doi: 10.1021/acs.jpcb.9b04729.
- [35] T.L. Toulias and C.P. Kitsos. Fitting the Michaelis–Menten model. *Journal of Computational and Applied Mathematics*, 296:303–319, 2016. doi: 10.1016/j.cam.2015.10.004.
- [36] S. Masroor, M.G.J. van Dongen, R. Alvarez-Jimenez, K. Burggraaf, L.A. Peletier, and M.A. Peletier. Mathematical modeling of the glucagon challenge test. *Journal of Pharmacokinetics and Pharmacodynamics*, pages 553–564, 2019. doi: 10.1007/s10928-019-09655-2.
- [37] P. Deepraj and P. Karthika. Aspergillosis: An Overview. *International Journal of Pharmaceutical Sciences and Research*, 9(12)(2):5032–5049, 2018. doi: 10.13040/IJPSR.0975-8232.9(12).5032-49.
- [38] J.P. Latgé. Aspergillus fumigatus and Aspergillosis. *Clinical Microbiology Reviews*, 12(2):310–350, 1999. doi: 10.1128/cmr.12.2.310.
- [39] A. Margalit and K. Kavanagh. The innate immune response to Aspergillus fumigatus at the alveolar surface. *FEMS microbiology reviews*, 39(5):670–687, 2015. doi: 10.1093/femsre/fuv018.
- [40] J.P. Latgé and G. Chamilos. Aspergillus fumigatus and aspergillosis in 2019. *Clinical microbiology reviews*, 33(1), 2019. doi: 10.1128/CMR.00140-18.
- [41] I. Hadrich, F. Makni, S. Neji, S. Abbes, F. Cheikhrouhou, H. Trabelsi, H. Sellami, and A. Ayadi. Invasive Aspergillosis: Resistance to Antifungal Drugs. *Mycopathologia*, 174(2):131–141, 2012. doi: 10.1007/s11046-012-9526-y.
- [42] J.M. Rybak, J.R. Fortwendel, and P.D. Rogers. Emerging threat of triazole-resistant aspergillus fumigatus. *Journal of Antimicrobial Chemotherapy*, 74(4):835–842, 2019. doi: 10.1093/jac/dky517.
- [43] P.P.A. Lestrade, J.F. Meis, W.J.G. Melchers, and P.E. Verweij. Triazole resistance in aspergillus fumigatus: recent insights and challenges for patient management. *Clinical Microbiology and Infection*, 25(7):799–806, 2019. doi: 10.1016/j.cmi.2018.11.027.
- [44] C.D. Lauruschkat, H. Einsele, and J. Loeffler. Immunomodulation as a therapy for Aspergillus infection: Current status and future perspectives. *Journal of fungi*, 4(4):1–17, 2018. doi: 10.3390/jof4040137.
- [45] J. Piñero, L.I. Furlong, and F. Sanz. In silico models in drug development: where we are. *Current Opinion in Pharmacology*, 42:111 – 121, 2018. doi: <https://doi.org/10.1016/j.coph.2018.08.007>.

- [46] R. Colquitt, D. Colquhoun, and R. Thiele. In silico modeling of physiologic systems. *Best practice & research. Clinical anaesthesiology*, 25:499–510, 2011. doi: 10.1016/j.bpa.2011.08.006.
- [47] A.K. Zaas, G. Liao, J.W Chien, C Weinberg, D Shore, S.S. Giles, K.A. Marr, J. Usuka, L.H. Burch, L. Perera, J.R. Perfect, G. Peltz, and D.A. Schwartz. Plasminogen alleles influence susceptibility to invasive aspergillosis. *PLoS Genetics*, 4 (6):1000101, 2008. doi: 10.1371/journal.pgen.1000101.
- [48] J. Thykaer, M.R. Andersen, and S.E. Baker. Essential pathway identification: From in silico analysis to potential antifungal targets in *Aspergillus fumigatus*. *Medical Mycology*, 47(SUPPL. 1):S80–7, 2009. doi: 10.1080/13693780802455305.
- [49] R. Thakur and J. Shankar. In silico identification of potential peptides or allergen shot candidates against *aspergillus fumigatus*. *BioResearch Open Access*, 5(1): 330–341, 2016. doi: 10.1089/biores.2016.0035.
- [50] R. Chaudhary, M. Balhara, D.K. Jangir, M. Dangi, M. Dangi, and A.K. Chhillar. In Silico Protein Interaction Network Analysis of Virulence Proteins Associated with Invasive Aspergillosis for Drug Discovery. *Current Topics in Medicinal Chemistry*, 19(2):146–155, 2018. doi: 10.2174/1568026619666181120150633.
- [51] N. Curty, P.H. Kubitschek-Barreira, G.W. Neves, D. Gomes, L. Pizzatti, E. Abdellahy, G.H.M.F. Souza, and L.M. Lopes-Bezerra. Discovering the infectome of human endothelial cells challenged with *Aspergillus fumigatus* applying a mass spectrometry label-free approach. *Journal of Proteomics*, 97:126–140, 2014. doi: 10.1016/j.jprot.2013.07.003.
- [52] R.J. Tanaka, N.J. Boon, K. Vrcelj, A. Nguyen, C. Vinci, D. Armstrong-James, and E. Bignell. In silico modeling of spore inhalation reveals fungal persistence following low dose exposure. *Scientific Reports*, 5:13958, 2015. doi: 10.1038/srep13958.
- [53] M. Oremland, K.R. Michels, A.M. Bettina, C. Lawrence, B. Mehrad, and R. Laubenbacher. A computational model of invasive aspergillosis in the lung and the role of iron. *BMC Systems Biology*, 10(1):34, 2016. doi: 10.1186/s12918-016-0275-2.
- [54] S. Kasahara, A. Jhingran, S. Dhingra, A. Salem, R. Cramer, and T. Hohl. Gm-csf signaling regulates neutrophil antifungal activity and the oxidative burst during respiratory fungal challenge. *Journal of Infectious Diseases*, 213:jiw054, 2016. doi: 10.1093/infdis/jiw054.
- [55] A. Jhingran, K.B. Mar, D.K. Kumashika, S.E. Knoblaugh, L.Y. Ngo, B.H. Segal, Y. Iwakura, C.A. Lowell, J.A. Hamerman, X. Lin, and T.M. Hohl. Tracing conidial fate and measuring host cell antifungal activity using a reporter of microbial viability in the lung. *Cell Reports*, 2(6):1762 – 1773, 2012. doi: 10.1016/j.celrep.2012.10.026.

- [56] S. Kasahara, A. Jhingran, S. Dhingra, A. Salem, R.A. Cramer, and T.M. Hohl. Role of Granulocyte-Macrophage Colony-Stimulating Factor Signaling in Regulating Neutrophil Antifungal Activity and the Oxidative Burst During Respiratory Fungal Challenge. *The Journal of Infectious Diseases*, 213(8):1289–1298, 2016. doi: 10.1093/infdis/jiw054.
- [57] A. Jhingran, S. Kasahara, K.M. Shepardson, B.A. Fallert Junecko, L.J. Heung, D.K. Kumasa, S.E. Knoblaugh, X. Lin, B.I. Kazmierczak, T.A. Reinhart, R.A. Cramer, and T.M. Hohl. Compartment-specific and sequential role of myd88 and card9 in chemokine induction and innate defense during respiratory fungal infection. *PLOS Pathogens*, 11(1):1–22, 2015. doi: 10.1371/journal.ppat.1004589.
- [58] E.E. Rosowski, N. Raffa, B.P. Knox, N. Golenberg, N.P. Keller, and A. Huttenlocher. Macrophages inhibit aspergillus fumigatus germination and neutrophil-mediated fungal killing. *PLOS Pathogens*, 14(8):1–28, 2018. doi: 10.1371/journal.ppat.1007229.
- [59] S. Kasahara, A. Jhingran, S. Dhingra, A. Salem, R.A. Cramer, and T.M. Hohl. Role of granulocyte-macrophage colony-stimulating factor signaling in regulating neutrophil antifungal activity and the oxidative burst during respiratory fungal challenge. *The Journal of infectious diseases*, 213(8):1289–1298, 2016. doi: 10.1093/infdis/jiw054.
- [60] A. Jhingran, K. Mar, D. Kumasa, S. Knoblaugh, L. Ngo, B. Segal, Y. Iwakura, C. Lowell, J. Hamerman, X. Lin, and T. Hohl. Tracing conidial fate and measuring host cell antifungal activity using a reporter of microbial viability in the lung. *Cell reports*, 2, 2012. doi: 10.1016/j.celrep.2012.10.026.
- [61] P. Lecca. *Model Identifiability*, chapter 3, pages 37–48. Springer International Publishing, Cham, 2020. ISBN 9783030412555.
- [62] M. Ashyraliyev, Y. Fomekong-Nanfack, J.A. Kaandorp, and J.G. Blom. Systems biology: Parameter estimation for biochemical models. *FEBS Journal*, 276(4): 886–902, 2009. doi: 10.1111/j.1742-4658.2008.06844.x.
- [63] A. Villaverde, A. Barreiro, and A. Papachristodoulou. Structural identifiability of dynamic systems biology models. *PLOS Comput Biol*, 12(10), 2016. doi: 10.1371/journal.pcbi.1005153.
- [64] J Schaber and Edda Klipp. Model-based inference of biochemical parameters and dynamic properties of microbial signal transduction networks. *Current Opinion in Biotechnology*, 22(1):109 – 116, 2011. doi: 10.1016/j.copbio.2010.09.014. Analytical biotechnology.
- [65] O. Chiş, J.R. Banga, and E. Balsa-Canto. GenSSI: a software toolbox for structural identifiability analysis of biological models. *Bioinformatics*, 27(18):2610–2611, 2011. doi: 10.1093/bioinformatics/btr431.
- [66] R. Li, M.A. Henson, and M.J. Kurtz. Selection of model parameters for off-line parameter estimation. *IEEE Transactions on Control Systems Technology*, 12(3): 402–412, 2004. doi: 10.1109/TCST.2004.824799.

- [67] B. Kim and J.H. Lee. Parameter subset selection and biased estimation for a class of ill-conditioned estimation problems. *Journal of Process Control*, 81:65–75, 2019. doi: 10.1016/j.jprocont.2019.05.015.
- [68] A. Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009. doi: 10.1093/bioinformatics/btp358.
- [69] M. Anguelova, J. Karlsson, and M. Jirstrand. Minimal output sets for identifiability. *Mathematical Biosciences*, 239(1):139 – 153, 2012. doi: 10.1016/j.mbs.2012.04.005.
- [70] M. Rodriguez-Fernandez, P. Mendes, and J.R. Banga. A hybrid approach for efficient and robust parameter estimation in biochemical pathways. *Biosystems*, 83(2):248 – 265, 2006. doi: 10.1016/j.biosystems.2005.06.016. 5th International Conference on Systems Biology.
- [71] E. Balsa-Canto, M. Peifer, J.R. Banga, J. Timmer, and C. Fleck. Hybrid optimization method with general switching strategy for parameter estimation. *BMC Systems Biology*, 2(1):26, 2008. doi: 10.1186/1752-0509-2-26.
- [72] J.A. Egea, E. Balsa-Canto, M.S.G. García, and J.R. Banga. Dynamic optimization of nonlinear processes with an enhanced scatter search method. *Industrial and Engineering Chemistry Research*, 48(9):4388–4401, 2009. doi: <http://dx.doi.org/10.1021/ie801717t>.
- [73] A. Gábor and J.R. Banga. Robust and efficient parameter estimation in dynamic models of biological systems. *BMC systems biology*, 9(1):74, 2015. doi: 10.1186/s12918-015-0219-2.
- [74] D.R. Penas, P. González, J.A. Egea, R. Doallo, and J.R. Banga. Parameter estimation in large-scale systems biology models: a parallel and self-adaptive cooperative strategy. *BMC bioinformatics*, 18(1):52, 2017. doi: 10.1186/s12859-016-1452-4.

Appendix A: Algorithm Implementation

The parameter estimation algorithms was implemented in MATLAB and is made available from: https://github.ic.ac.uk/jz2816/FinalReport_Code.

Appendix B: Supplementary for Genetic Algorithm Results

This section is the full matrix plot for population at generation number 1, 3 and 30. In Each figure: the diagonal plots are parameter distributions with corresponding parameters labelled. The lower triangular plots are pairwise scatter plot between two parameters (.) and the pairwise nominal parameter values (x). The parameter pairs can be read from the position of the subplots, where the column and the row indicate the parameter being plotted on x and y axis, respectively. The upper triangle panels are the correlation coefficients (R) corresponding to the parameter pairs of the scatter plots at the opposite side along the diagonal. The intensity of the coloured circles in upper triangular area is proportional to the magnitude of R. These plots together with figure 11, which is full population matrix plot at generation number 200, demonstrated genetic algorithm converges to local minimum.

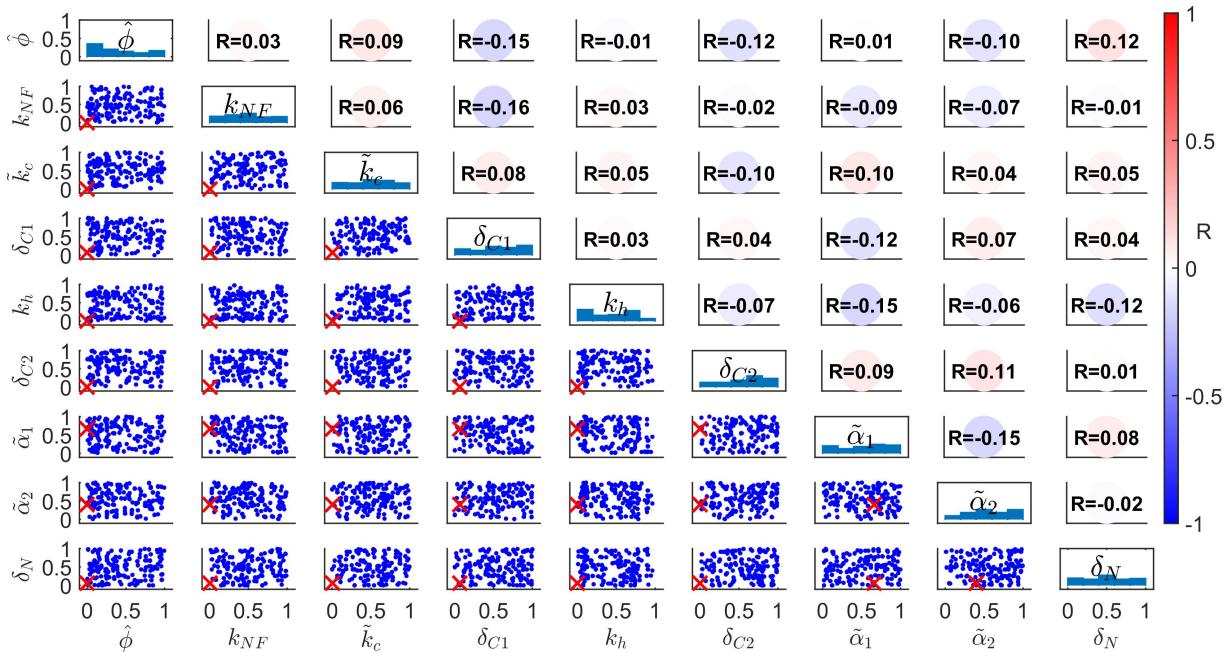


Figure 19: Generation number 1

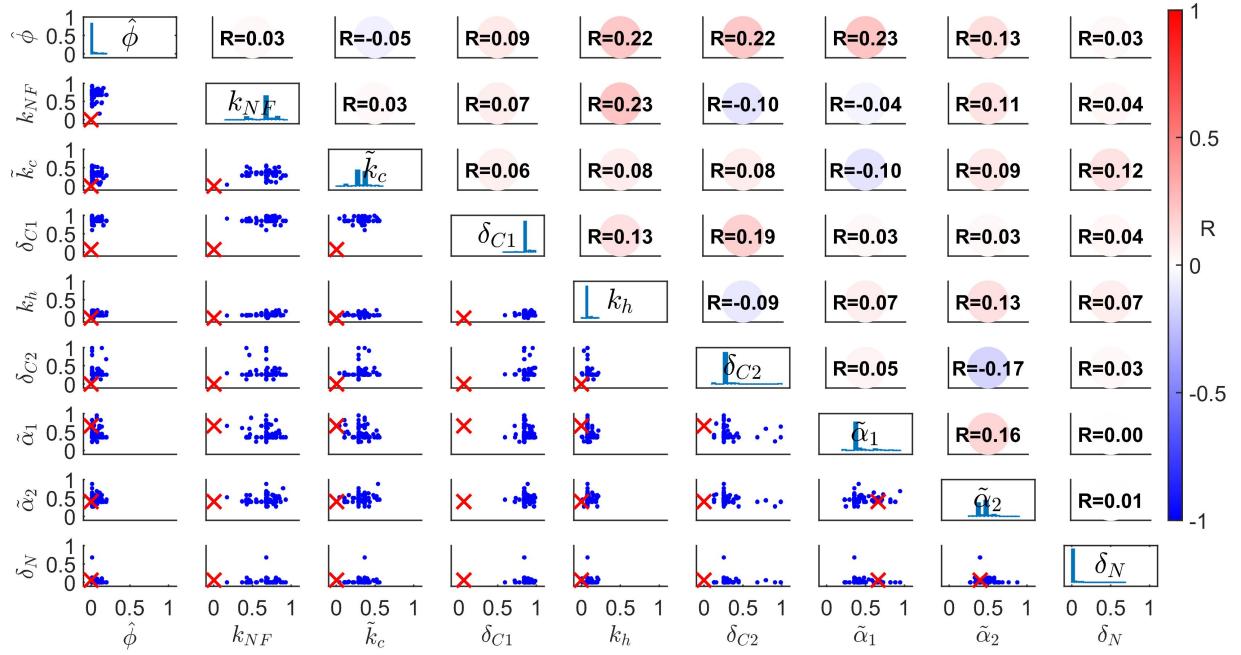


Figure 20: Generation number 15

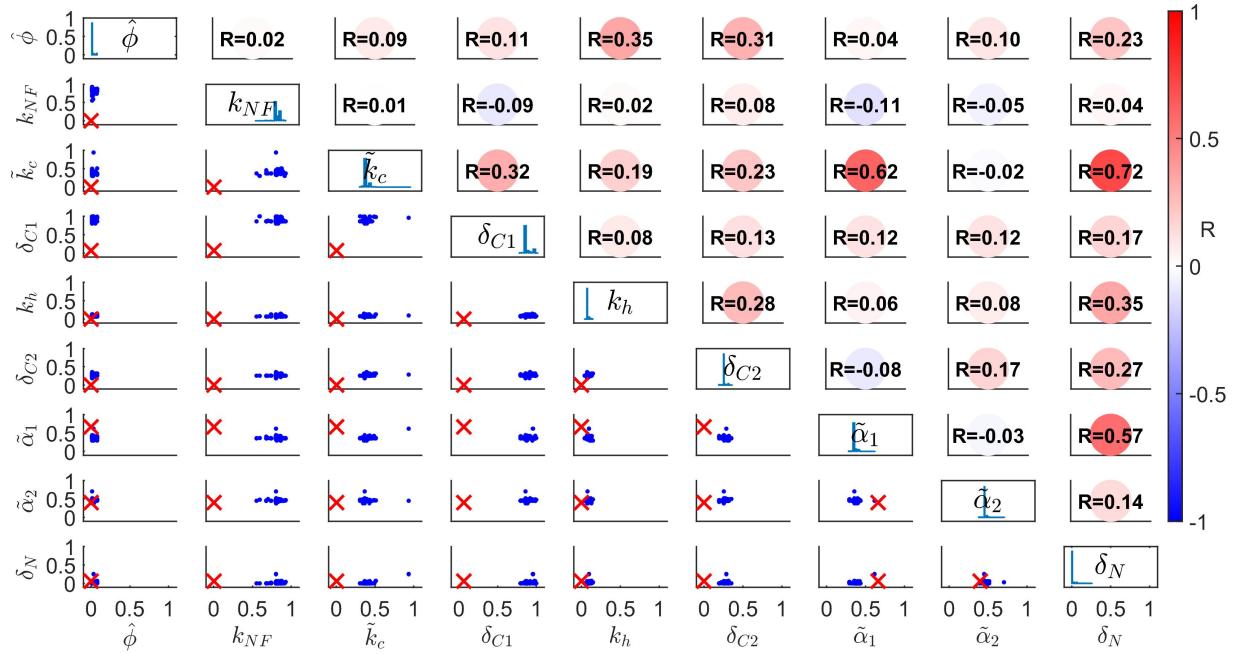


Figure 21: Generation number 30