Lappeenranta University of Technology
School of Engineering Science
Computational Engineering and Technical Physics
Technomathematics

Amadi, Chinalu Miracle

AGENT-BASED MODELING FOR MOSQUITO CONTROL

Master's Thesis

Examiners: Professor Heikki Haario
Professor Tuomo Kauranne
Supervisor: Professor Heikki Haario

ABSTRACT

Lappeenranta University of Technology
School of Engineering Science
Computational Engineering and Technical Physics
Technomathematics

Amadi, Chinalu Miracle

Agent-Based Modeling for Mosquito Control

Master's Thesis

2018

82 pages, 19 figures, 8 tables, and 1 appendice.

Examiners: Professor Heikki Haario
Professor Tuomo Kauranne

Keywords: Malaria control, Insecticidal treated nets, Agent-based modeling, Markov chain Monte Carlo, Model calibration.

Malaria infection remains a major health challenge in many parts of the world, especially in sub-Saharan Africa. Although intervention strategies abound, the efficiency of control measures emanates from understanding the host-seeking behaviors of the malaria vector species. Discrete agent-based modeling was employed to demonstrate the protective efficacy of Insecticidal treated nets on *Anopheles gambiae* and *Anopheles arabiensis* by considering the treatment measure alongside the control case; with just a physical net barrier. The aforementioned modeling approach is indeed suitable in demonstrating the dissimilar host-seeking behavioral tendencies of *An. gambiae* and *An. arabiensis* when confronted with Insecticidal treated nets. The models are calibrated against several real experimental hut trial datasets by employing the adaptive Markov chain Monte Carlo. Both the control and treatment case models are able to replicate the datasets considered, virtually within the reported error bounds. Moreover, the datasets employed are able to properly identify the sampled parameters in both cases, and as such, reveal relatively low uncertainties in these model parameters.

PREFACE

My heart is overwhelmed with joy, as I accord my profound gratitude to the Almighty God for enabling me complete this study to His glory. I must express my sincere gratitude to a lot of people who have contributed in diverse ways to make this report a success. My deepest gratitude is to my supervisor, Professor Heikki Haario. I have been incredibly lucky to have a supervisor who gave me both the chance to explore new ideas on my own and guidance when I lost my way, despite his busy schedules. I must say that working with you is really rewarding and gratifying. My heartfelt thanks goes to Mrs. Anna Shcherbacheva for being generous with her constructive guidelines and her codes which made it easy for me to simulate the models, it would not have been easier otherwise. To my lecturers at LUT Finland, I say a big thanks to you all for a job well done. Most importantly, none of this would have been possible without the love and support of my family and my husband, Rev. Emeka Godson, who have been a constant source of inspiration, hope and joy to me. Finally, let me seize this opportunity to acknowledge the efforts of the administrative staff of LUT, Finland for offering me the opportunity and support to make this work a huge success.

Lappeenranta, April 20, 2018

Amadi, Chinalu Miracle

CONTENTS

ABBREVIATIONS AND SYMBOLS

ITN	Insecticidal treated net
LLIN	Long-lasting insecticidal net
N_m	Number of mosquitoes
$\mathbf{d}_{\text{old}}$	A vector of old mosquito distance
$\mathbf{d}_{\text{new}}$	A vector of new mosquito distance
$\mathbf{x}_{\text{cord}}$	A vector of X coordinates
$\mathbf{y}_{\text{cord}}$	A vector of Y coordinates
h_s	Hut size
d_p	Net width
$\sim$	Not
$\&$	And
$\vert$	Or

1 INTRODUCTION

In this section, a motivation for undertaking this study is given. The role of mosquitoes in the transmission of malaria and the relevance of long-lasting insecticidal nets (LLINs) in the reduction of such transmission is shown. Then we introduce an agent-based approach for modeling the host-seeking behavior of mosquitoes. The problem of interest is further stated in this section, and the purpose of the study is given together with the expected achievements in the present work. Finally, the relevance of the study to the society is discussed.

1.1 Background of Study

1.1.1 Global Burden Posed by Malaria

Malaria has been a health threat to humankind and is still a significant threat to virtually half of the world's population especially in sub-Saharan Africa [1]. It is estimated that there were about 212 million cases of malaria worldwide, leading to about 429,000 deaths just in 2015. In this period, 92% of the estimated deaths occurred in Africa [2]. The groups most affected by this pandemic are children below the age of five and pregnant women [3]. Hence, it is imperative that active and continuous research be conducted on malaria, since the groups most vulnerable to this disease are widely of great economic importance. Given that the socio-economic impact of malaria is so high that it contributes highly to poverty and underdevelopment, it follows that, malaria research is more than just a study in public health, but also a great contribution to the socio-economic improvement of Africa and the world at large.

1.1.2 Role of Mosquitoes in Malaria Transmission

A protozoan parasite, known as Plasmodium, is the pathogen responsible for causing malaria. In humans, Plasmodium falciparum, Plasmodium vivax, Plasmodium malariae and Plasmodium ovale are the four species of Plasmodium known to cause malaria. Among these four parasite species, the predominant source of infection in Africa is Plasmodium falciparum. It is responsible for both 80 percent of all recorded malaria incidents and 90 percent of malaria related deaths in Africa [4]. Malaria is transmitted during the blood feeding of infectious female Anopheles mosquitoes that are in need of blood meals

to nurture their eggs. Thus, the complete life cycle of malaria parasites involves two hosts; humans and female Anopheles mosquitoes. First, the female Anopheles mosquito ingests the parasites (gametocytes) from a person infected with malaria during blood feeding which it needs to nurture its eggs. Thus, the parasites develop and reproduce inside the mosquito gut after-which they are transferred into the mosquito salivary glands. Upon bitting the human, the parasites (sporozoites) contained in their salivary gland are injected into the human bloodstream from where they invade the liver cells. In the liver stage, each sporozoite multiplies and develops into thousands of schizonts, which rupture and release merozoites and thereafter, infest the red blood cells. After maturity, the merozoites can either develop into gametocyte or invade and destroy other red blood cells and this makes the infected person to experience malaria symptoms [5]. The schematic depiction of the malaria life cycle is given in Figure 1.

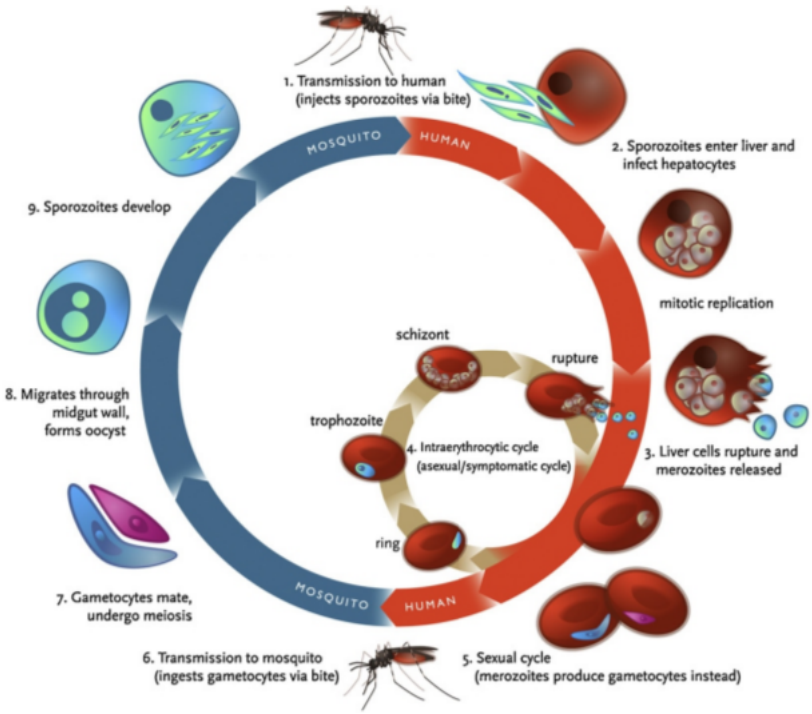


Figure 1. Malaria Life Cycle (Source:[6]).

From the above figure, it can be deduced that eliminating mosquitoes from the cycle will eradicate Plasmodium parasites and malaria from the human populace.

1.1.3 Host-Seeking, Resting and Feeding Traits of Mosquitoes

It is a necessity to understand the characteristics of mosquitoes under certain conditions before planning for mosquito control strategies. Mosquito species exhibit a wide variety of differences in their resting and feeding behaviour, alongside their host preferences. Mosquitoes can be classified according to their behavioral traits. In the search for meals during a gonotrophic cycle, mosquitoes exhibit an assortment of characteristic traits.

First, we consider their meal preference. Female mosquitoes have the need to feed on blood to develop eggs in their ovaries. The blood meal does not necessarily have to come from human sources. Some mosquitoes can have their blood meal on other vertebrates such as cows as an alternative to humans. Mosquitoes that prefer feeding on humans alone are known as anthropophilic for example, *Anopheles gambiae* [7] while the species that have attraction for other vertebrates are known as zoophilic for instance, *Anopheles arabiensis* [8]. ITNs/LLINs impact the feeding cycle of each kind of species differently. In the case of anthropophilic species, it lengthens greatly the time required in obtaining a blood meal since they feed primarily on humans. Consequently, either the time required to complete a gonotrophic cycle is lengthened, or the mosquito dies of starvation. On the other hand, zoophilic species are forced to take a larger proportion of their blood meals from non-human sources. In either case, the number of humans bitten by mosquitoes, as they sleep at night is reduced and the risk of malaria infection is decreased.

Considering their preference of resting place after a successful blood meal in order to allow digestion of blood, there are two families of mosquitoes in this respect; endophilic and exophilic mosquitoes. Mosquito species that rest indoors, inside a human abode after a successful blood meal are endophilic mosquitoes *An. gambiae*, for instance while exophilic mosquitoes pass this period of rest outside human dwellings, as in the case of *An. arabiensis* [9].

Aside from resting behaviour, mosquitoes can be classified according to their preference of biting location, whether they prefer to bite outdoors or indoors. Certain species such as *An. gambiae* and *An. funestus* prefer feeding indoors and are described as endophagic whereas, others such as *An. arabienesis* are described as exophagic since they primarily feed outdoors [10].

A shared feature of mosquitoes which is of utmost importance to be studied in this work, is their ability to detect the human host for blood meal by making use of the olfactory cues that are given off by human. Mosquitoes are able to sense from a long distance, the carbon

dioxide (CO_2) exhaled by humans using their maxillary palps and other chemical odours emitted from the human body by using their antennae [11]. As stated by [12], many substances contained in the human sweat, were identified as attractive for mosquitoes, such compounds include: lactic acid, phenol, ammonia and nonanol. Also, mosquitoes are able to use the heat sensors around their mouth-parts to sense human from a distance and to discern the warmth of a human's body. They can also recognize movement, colors, shapes and patterns via vision but they are not able to identify the human host from a distance greater than 80 meters [13].

Furthermore, we focus on *An. gambiae* and *An. arabiensis* mosquitoes since in any case, they are still susceptible to indoor interventions such as the ITNs, which are the main control measure considered in the study.

1.1.4 Overview of Long-Lasting Insecticidal Nets

The control of mosquitoes that are capable of transmitting malaria parasites is an essential strategy of malaria prevention. There are many strategies available for mosquito control, among which indoor residual spraying and insecticide-treated nets are the most effective methods [14]. This emanates from the fact that the principal malaria vectors, primarily feed indoors at night when people are asleep [15]. Therefore, the use of nets provide an effective way of protecting humans from such mosquito bites when they are asleep.

Nets can either be treated with insecticides or used untreated. Untreated nets only provide a physical roadblock against mosquitoes which attempt to bite humans. However, insecticidal nets provide protection through chemical action on mosquitoes as well as presenting a material barrier against the malaria vectors. Thus, the use of insecticide-treated nets is more advantageous than the use of untreated nets. This is because, treated nets kill some mosquitoes that land on them and prevent them from attacking an unprotected person in the household. Also, nets treated with repellents prevent some mosquitoes from entering the house to bite humans.

The use of LLINs has been recommended by the World Health Organisation because of their durability and also due to the fact that they do not require regular re-treatment unlike the way other conventional insecticide-treated nets do, even though they should also be replaced when their active period has expired. The aforementioned net can last for 3 years and can still be active even after 20 standard washing because they have quality controlled insecticide that is being incorporated in their fibres [16].

However, irrespective of the large scale distribution of LLINs in many malaria endemic countries, there is a wide variation in the availability and the degree of use (coverage) of LLINs in different households. Studies have shown that most households that own LLINs have either one or two LLINs irrespective of their household size which is not sufficient [17]. Thus, in order to effectively control malaria, it is necessary that sufficient nets are continually made available to protect household members so that over time, coverage can approach 100%.

1.1.5 Agent-Based Modeling

The dynamics of malaria transmission is complex, with many factors influencing its temporal and spatial patterns. Several mathematical models have been formulated over time, to investigate the transmission dynamics of malaria. Although they proffer illuminating headway into the transmission system, they neglect the influence of some cause-effect relationships and are only applicable for modeling the collective behavior of a group. These mathematical models rely on a compartmental structure in which humans or mosquitoes transit from one compartment to the next at fixed parameter rates and are assumed to have uniform characteristics within each compartment. Thus with such kind of models, it is only possible to update the status of groups but the specific individual characteristics cannot be tracked. In this work, however, we shall use a different model structure known as the agent-based modeling, in order to give more insight for studying certain malaria transmission dynamics that are often isolated in compartmental models.

Agent-based modeling (ABM) is a computational modeling approach which represents individual entities in a complex system as discrete agents that interact autonomously in a simulated space and time by following a set of predefined rules, in order to achieve their objectives [18]. The agents for this study are autonomous (self directed) mosquitoes and human that are simulated to reside in a 2 dimensional space (the hut) and are driven by external factors of which we assume four major driving factors: the attraction of odour emitted from the host and sensed by mosquitoes, repulsion by a physical net barrier posed by an untreated net, repulsion effect of a treated net and the poisoning effect of the treated net. Using this modeling approach, relevant characteristics for individual mosquitoes are tracked. Such characteristics include position, blood meal status, quantity of accumulated chemicals and mortality status. Each of these attributes are updated at each simulation time-step as the mosquitoes move randomly throughout the simulated transmission domain. Thus, this modeling approach is discrete since it handles sequence of events that occur at a particular instant in time and also describes the behaviour of each individ-

ual entity. Therefore discrete ABM is employed in this work to assess the impacts of ITNs/LLINs in controlling mosquitoes.

1.2 Statement of Problem

Variants of host-seeking behaviors of mosquitoes of different species have long been studied by many researchers in their various capacities and still, it is a major subject of concern, owing to the fact that mosquitoes of different species exhibit various behavioral traits with respect to the control measures to which they are confronted with [19], [20], [21], [22], [23]. While learning about the outcomes emanating from researches on mosquito host-seeking behavior, it is of paramount importance to develop robust and ideal agent-based models and numerical frameworks that capture the essential host-seeking behaviors of mosquitoes in different conditions (precisely at hut-level with a treated and an untreated net) while parsimoniously employing parameters that are pivotal to the basic responses and thereby validate these models such that they replicate multiple real experimental data while taking data uncertainty and the uncertainty posed by the randomness inherent in the aforementioned modeling approach into cognizance.

1.3 Research Objectives

The specific objectives emerging from the problem statement discussed above include:

- To give highlights to the existing models that describe the host-seeking behavior of mosquitoes, in a bid of making possible restructuring, where necessary.

- To simulate a hut-level experiment in order to study the impact of ITNs/LLINs on the host-seeking behavior of *An. gambiae* and *An. arabiensis* .

- To study the variability of the outputs of the hut experiment in the control case alongside the sensitivity of its model parameters.

- To calibrate both the control case and treatment case, hut experiment models against several real measurements while taking data uncertainty into consideration.

1.4 Significance of Study

- The findings of this study will act as a support for further research on malaria vector control.
- The findings of this study will help the society in general, to be aware of the necessity for sufficient coverage with ITNs/LLINs.
- This study will be of tremendous help to the governments as it will engender them to take into consideration the importance of ITNs/LLINs in controlling malaria while planning for malaria control strategies.

1.5 Scope of Study

In this study, we model the host-seeking behaviour of *An. gambiae* and *An. arabiensis* in a hut-level experiment with two indoor intervention measures (treated and untreated bed nets). The control case model is calibrated with five real sets of data while the treatment case model is calibrated against four real experimental data.

1.6 Structure of the Report

This report comprises five sections. Section 1 is a general introduction to the report, and has six subsections. Section 1.1 gives the motivation of this work, and general background information of the problem including information about the vector species under study. The specific question we wish to answer and what we hope to achieve at the end of the study are given in Sections 1.2 and 1.3. The anticipated relevance of this study to the society is discussed in Section 1.4. Section 1.5 states the scope of this work. Finally, this section (Section 1.6) gives a roadmap to the arrangement of the entire report.

Section 2 gives an overview of Markov chain Monte Carlo (MCMC) methods for parameter estimation and how its features are employed in the work and it comprises 3 subsections. The concept of Bayesian parameter estimation as it links to MCMC is given in Section 2.1. In Section 2.2, the algorithmic descriptions of some MCMC approaches, alongside there modifications and extensions are discussed. MCMC convergence diagnostics methods are given in Section 2.3.

The description of the models implemented in this work is given in Section 3. Its first four subsections (Sections 3.1 to 3.4), describe the mosquito host-seeking behavior and its mortality models. Section 3.5 presents the model algorithm and the description of the parameters employed in the models is given in Section 3.6. Finally, the assumptions of the models are given in the last section (Section 3.7).

Section 4 presents the results of various analysis and model calibrations done. The uncertainty quantification of the model outputs in the control case and the sensitivity analysis of its model parameters are given in Sections 4.1 and 4.2 and the calibration of the models against real measurements is featured in Section 4.3.

Finally, Section 5 discusses the report in general. Section 5.1 gives an overview of what has been done. The conclusion based on the results obtained in Section 4, is given in Section 5.2. The recommendations and relevant future works are given in Section 5.3.

2 MARKOV CHAIN MONTE CARLO METHOD FOR PARAMETER ESTIMATION

In this section, the introduction and explanation of the basic role of Markov chain Monte Carlo (MCMC) in parameter estimation, especially in cases of high dimensional problems, as in the case of our study, is given. We briefly explain the concepts of Metropolis algorithm, Metropolis-Hastings algorithm and Adaptive MCMC with its improved version. Finally, the common diagnostics that can be done in order to access the convergence of these algorithms are discussed.

2.1 Bayesian Parameter Estimation

In statistical analysis of mathematical models we aim not only to estimate the optimal values of unknown parameters, but the estimation of the distribution of these parameters is also of paramount interest. This introduces us to the Bayesian parameter estimation approach where the unknown parameters, $\boldsymbol{\theta}$, in the models are thought to be random variables from certain distributions. Hence, the first goal is to find the posterior distribution $\pi(\boldsymbol{\theta}|\mathbf{y})$ of the parameters , which gives the probability density for the values of $\boldsymbol{\theta}$, given a measured data $\mathbf{y}$, parametrised as

$$\pi(\boldsymbol{\theta}|\mathbf{y}) = \frac{l(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathbf{y})}, \tag{1}$$

where $l(\mathbf{y}|\boldsymbol{\theta})$ is the likelihood which contains the information from the sample data and gives the probability density of observing data $\mathbf{y}$, given the parameter value, $\boldsymbol{\theta}$. The prior distribution $p(\boldsymbol{\theta})$ represents the previous knowledge about $\boldsymbol{\theta}$ before taking the empirical evidence into account. In most cases, the uninformative flat prior; which expresses a vague knowledge of $\boldsymbol{\theta}$, is used and hence $p(\boldsymbol{\theta}) = 1$. The posterior distribution, $\pi(\boldsymbol{\theta}|\mathbf{y})$ reveals the level of uncertainty associated with the parameter set, $\boldsymbol{\theta}$ after taking both the prior information and the data into consideration. The normalizing constant; $p(\mathbf{y}) = \int_{\Theta} l(\mathbf{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$ ensures that the posterior $\pi(\boldsymbol{\theta}|\mathbf{y})$ integrates to 1. However, working with the posterior density is quite challenging, especially in high dimensional cases because one has to integrate over the parameter space in order to calculate the normalising constant for the posterior density. Thus MCMC is introduced so that the posterior distribution can be evaluated without explicitly computing the normalising constant, $p(\mathbf{y})$.

2.2 MCMC Methods

The MCMC methods are based on generating a sequence of N-random samples $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_N)$, from a proposal distribution and then correcting those draws so that their distribution asymptotically approaches the target posterior distribution as N increases [24]. The Monte Carlo term is used to describe methods that are based on generating random numbers and subsequently averaging over the simulations while the Markov chain component is based on the fact that the sequence of samples are generated such that each new point $\boldsymbol{\theta}_{n+1}$ depends only on the previous point $\boldsymbol{\theta}_n$. The MCMC methods take into account all the uncertainties in the data and yield a sample of the whole posterior distribution of parameters instead of only a point estimate, as in the least squares. Some MCMC methods such as Metropolis algorithm, Metropolis–Hastings algorithm, adaptive MCMC and delayed rejection adaptive Metropolis algorithms are hence discussed briefly.

2.2.1 Metropolis Algorithm

The Metropolis algorithm is one of the commonly used MCMC algorithm [25]. It uses the accept/reject procedure which works by drawing candidate parameter values from a symmetric proposal distribution, $q(\cdot|\boldsymbol{\theta})$, which are either accepted or rejected based on some probability. In Algorithmic form, the method is described in Algorithm 1 below.

Algorithm 1 Metropolis Algorithm

1. Initialise the parameter $\boldsymbol{\theta}_0$ and set $n = 0$;
2. From a suitable proposal distribution, $q(\cdot|\boldsymbol{\theta}_n)$, sample a new candidate point $\hat{\boldsymbol{\theta}} \sim q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}_n)$ that depend on the previous point on the chain;
3. Compute an acceptance probability

$$\alpha_a(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}_n) = \min\left(1, \frac{\pi(\hat{\boldsymbol{\theta}})}{\pi(\boldsymbol{\theta}_n)}\right) \tag{2}$$

4. Sample a random number, u from a uniform distribution $\mathrm{U}(0, 1)$;
5. Accept $\hat{\boldsymbol{\theta}}$, if $u < \alpha_a(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}_n)$ and set $\boldsymbol{\theta}_{n+1} = \hat{\boldsymbol{\theta}}$; otherwise, reject $\hat{\boldsymbol{\theta}}$ and set $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n$;
6. Set $n = n + 1$ and repeat steps 2 through 6.

In the Metropolis algorithm, we only need to compute the ratios of the posterior distribution and in doing this, the problematic normalising constant cancels out. This is what MCMC does to handle multi-dimensional parameter estimation problems.

2.2.2 Metropolis–Hastings Algorithm

The Metropolis–Hastings algorithm is a simple extension of the Metropolis algorithm since the former assumes an asymmetric proposal distribution and its acceptance probability is slightly modified to account for asymmetry [26]. Thus, in Metropolis–Hastings algorithm, the probability density of moving from the current point to the proposed point is different from that of moving from the proposed point to the current point [26]. So, for two parameter values $\boldsymbol{\theta}_n$ and $\hat{\boldsymbol{\theta}}$ we have that $q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}_n) \neq q(\boldsymbol{\theta}_n|\hat{\boldsymbol{\theta}})$. Thus the probability of accepting the move from $\boldsymbol{\theta}_n$ to $\hat{\boldsymbol{\theta}}$ is given as

$$\alpha_a(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}) = \min\left(1, \frac{\pi(\hat{\boldsymbol{\theta}})\, q(\boldsymbol{\theta}_n|\hat{\boldsymbol{\theta}})}{\pi(\boldsymbol{\theta}_n)\, q(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta}_n)}\right). \tag{3}$$

Furthermore, a major concern in implementing the Metropolis algorithm and Metropolis–Hastings algorithms is choosing a proposal distribution. A paramount thing to consider in choosing the proposal distribution, is that it should be easy to sample from and it should be very close to the underlying target posterior distribution. Selecting an unsuitable proposal distribution can lead to inefficient implementation [27, see]. Mostly, a good guess for the proposal distribution if it is not known, is the Gaussian distribution with mean, $\boldsymbol{\theta}_n$ (current position) and a fixed covariance matrix. This is due to its theoretical and computational properties. Both Metropolis algorithm and Metropolis–Hastings algorithm require a manual tuning, which is laborious and so several adaptive MCMC algorithms have been developed to update the covariance matrix during the MCMC run.

2.2.3 Adaptive MCMC

A careful choice of a proposal distribution that matches the target distribution, is a necessity for the Metropolis algorithm to work efficiently. Choosing a Gaussian proposal distribution with mean $\boldsymbol{\theta}_n$ and a fixed covariance matrix as in the Metropolis algorithm approach does not always lead to efficient sampling. Thus the adaptive MCMC was developed in order to improve the proposal during the run using the information of the previously sampled points [28]. This is achieved by computing the empirical covariance matrix of the points sampled so far and using that as the proposal covariance matrix, thus making the algorithm non Markovian.

For the Adaptive Metropolis algorithm, the proposal is taken to be Gaussian, centered at

the current point and the proposal covariance matrix is an empirical covariance matrix which is computed using the information cumulated so far. To be more exact, assuming that we sampled points $(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{n-1})$ the new proposal distribution, $q_n(\cdot|\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{n-1})$ for the next candidate point, $\hat{\boldsymbol{\theta}}$ is the (asymptotically symmetric) Gaussian distribution with mean at the current point and the covariance matrix given as

$$\mathbf{C}_n = s_d \text{Cov}(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{n-1}) + s_d \omega \mathbf{I}_d, \tag{4}$$

where d is the dimension of $\boldsymbol{\theta}$, s_d is the scaling factor, $\omega > 0$ is the diagonal regularisation parameter that ensures the positivity of the covariance matrix and $\mathbf{I}_d$ is a $d \times d$ identity matrix. The candidate point $\hat{\boldsymbol{\theta}}$ is accepted with a probability, as given in Equation 2.

The vital thing to know about adaptation is how the covariance matrix of the proposal distribution depends on the chain history. Thus to start the adaptation process, we select an arbitrary strictly positive definite covariance $\mathbf{C}_0$, based on the prior knowledge. Next, we arbitrarily select a time index $n_0 > 0$ for the length of the initial non-adaptation period and define

$$\mathbf{C}_n = \begin{cases} \mathbf{C}_0, & n \leq n_0 \\ s_d \text{Cov}(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_{n-1}) + s_d \omega \mathbf{I}_d & n > n_0 \end{cases}, \tag{5}$$

where the covariance $\mathbf{C}_n$ may be viewed as a function of n variables from $\mathbb{R}^d$ whose values are in the uniformly positive definite matrices. Thus we have that the empirical covariance matrix determined by the points $(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_k) \in \mathbb{R}^d$ is given by

$$\text{Cov}(\boldsymbol{\theta}_0, \ldots, \boldsymbol{\theta}_k) = \frac{1}{k} \left(\sum_{i=0}^{k} \boldsymbol{\theta}_i \boldsymbol{\theta}_i^T - (k+1) \bar{\boldsymbol{\theta}}_k \bar{\boldsymbol{\theta}}_k^T \right), \tag{6}$$

where $\bar{\boldsymbol{\theta}}_k = \frac{1}{k+1} \sum_{i=0}^{k} \boldsymbol{\theta}_i$ is the empirical mean and the elements $\boldsymbol{\theta}_i \in \mathbb{R}^d$ are considered as column vectors [28]. Thus, substituting Equation 6 into Equation 5 for $n \geq n_0 + 1$, we obtain that the covariance matrix $\mathbf{C}_n$ satisfies the recursive formula

$$\mathbf{C}_{n+1} = \frac{n-1}{n} \mathbf{C}_n + \frac{s_d}{n} \left(n \bar{\boldsymbol{\theta}}_{n-1} \bar{\boldsymbol{\theta}}_{n-1}^T - (n+1) \bar{\boldsymbol{\theta}}_n \bar{\boldsymbol{\theta}}_n^T + \boldsymbol{\theta}_n \boldsymbol{\theta}_n^T + \omega \mathbf{I}_d \right). \tag{7}$$

This allows one to compute the covariance update without much computational cost since the mean $\bar{\boldsymbol{\theta}}_n$, also satisfies the recursive formula

$$\bar{\boldsymbol{\theta}}_n = \frac{n-1}{n} \bar{\boldsymbol{\theta}}_{n-1} + \frac{1}{n} \boldsymbol{\theta}_n. \tag{8}$$

The Adaptive MCMC approach is summarised in Algorithm 2 below.

Algorithm 2 Adaptive Metropolis

1. Initialise $\boldsymbol{\theta}_1$ and $\mathbf{C}_1$ and choose the chain length, N;
   While the number of iteration is less than the chain length:
2. Sample a new candidate point $\hat{\boldsymbol{\theta}} \sim \mathrm{N}(\boldsymbol{\theta}_n, \mathbf{C}_n)$;
3. Compute an acceptance probability

$$\alpha_a(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}_n) = \min\left(1, \frac{\pi(\hat{\boldsymbol{\theta}})}{\pi(\boldsymbol{\theta}_n)}\right); \tag{9}$$

4. Sample a random number, u from a uniform distribution $\mathrm{U}(0, 1)$;
5. Accept $\hat{\boldsymbol{\theta}}$, if $u < \alpha_a(\hat{\boldsymbol{\theta}}, \boldsymbol{\theta}_n)$ and set $\boldsymbol{\theta}_{n+1} = \hat{\boldsymbol{\theta}}$; otherwise, reject $\hat{\boldsymbol{\theta}}$ and set $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n$;
6. Update the covariance;

$$\mathbf{C}_{n+1} = s_d \mathrm{Cov}(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_n) + s_d \omega \mathbf{I}_d; \tag{10}$$

7. Set $n = n + 1$ and repeat steps 2 through 6.

There are different variants of implementing the adaptive Metropolis algorithm. The covariance matrix can be computed by the whole chain or by an increasing part of the chain [28]. Also, for proper and efficient adaptation, the covariance matrix can be updated after a certain interval of steps, instead of doing that at every step.

2.2.4 Delayed Rejection Adaptive Metropolis (DRAM)

More advanced adaptive MCMC exist such as the delayed rejection adaptive Metropolis (DRAM) algorithm [29]. This approach combines the delayed rejection (DR) method by [30] and the adaptive Metropolis Method. The basic idea of the DR is hence described.

Suppose that the current position of the chain is $\boldsymbol{\theta}_n$, a candidate point $\hat{\boldsymbol{\theta}}_1$ is generated from a proposal $q_1(\cdot|\boldsymbol{\theta}_n)$ and accepted with the probability

$$\begin{aligned} \alpha_1(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1) &= \min\left(1, \frac{\pi(\hat{\boldsymbol{\theta}}_1) q_1(\boldsymbol{\theta}_n|\hat{\boldsymbol{\theta}}_1)}{\pi(\boldsymbol{\theta}_n) q_1(\hat{\boldsymbol{\theta}}_1|\boldsymbol{\theta}_n)}\right), \\ &= \min\left(1, \frac{L_1}{G_1}\right), \end{aligned} \tag{11}$$

just as in the Metropolis–Hastings algorithm. If rejected, instead of repeating the previous value on the chain $\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n$ as we would do in Metropolis–Hastings algorithm, a second stage sampling attempt is made close to the current point using a different proposal, $q_2(\cdot|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)$ which depends on both the current point of the chain and on the point that was proposed and rejected. According to [31], the second stage proposal is accepted with a suitably modified acceptance probability, given as

$$\begin{aligned}\alpha_2(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2) &= \min\left(1, \frac{\pi(\hat{\boldsymbol{\theta}}_2)q_1(\hat{\boldsymbol{\theta}}_1|\hat{\boldsymbol{\theta}}_2)q_2(\boldsymbol{\theta}_n|\hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2)[1-\alpha_1(\hat{\boldsymbol{\theta}}_2, \hat{\boldsymbol{\theta}}_1)]}{\pi(\boldsymbol{\theta}_n)q_1(\hat{\boldsymbol{\theta}}_1|\boldsymbol{\theta}_n)q_2(\hat{\boldsymbol{\theta}}_2|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)[1-\alpha_1(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)]}\right), \\ &= \min\left(1, \frac{L_2}{G_2}\right)\end{aligned} \tag{12}$$

If the second stage is reached, it implies that $L_1 < G_1$. Thus in G_2, we replace, $\alpha_1(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)$ with L_1/G_1 and we have

$$\begin{aligned}\alpha_2(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2) &= \min\left(1, \frac{N_2}{G_1 q_2(\hat{\boldsymbol{\theta}}_2|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)\left[1-\frac{L_1}{G_1}\right]}\right), \\ &= \min\left(1, \frac{N_2}{q_2(\hat{\boldsymbol{\theta}}_2|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)[G_1 - L_1]}\right).\end{aligned}$$

Generally, the i-th stage of the Delayed Rejection algorithm works as follows. If a previously proposed candidate point $\hat{\boldsymbol{\theta}}_{i-1}$ is rejected, generate $\hat{\boldsymbol{\theta}}_i$ from $q_i(\cdot|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots)$ which is accepted with probability

$$\begin{aligned}\alpha_i(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_i) = \min\Bigg(1, \Bigg\{ &\frac{\pi(\hat{\boldsymbol{\theta}}_i)q_1(\hat{\boldsymbol{\theta}}_{i-1}|\hat{\boldsymbol{\theta}}_i)q_2(\hat{\boldsymbol{\theta}}_{i-2}|\hat{\boldsymbol{\theta}}_{i-1}, \hat{\boldsymbol{\theta}}_i)\cdots q_i(\boldsymbol{\theta}_n|\hat{\boldsymbol{\theta}}_{i-1}, \hat{\boldsymbol{\theta}}_i, \ldots)}{\pi(\boldsymbol{\theta}_n)q_1(\hat{\boldsymbol{\theta}}_1|\boldsymbol{\theta}_n)q_2(\hat{\boldsymbol{\theta}}_2|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)\cdots q_i(\hat{\boldsymbol{\theta}}_i|\hat{\boldsymbol{\theta}}_1, \boldsymbol{\theta}_n, \ldots)} \\ &\frac{\left[1-\alpha_1(\hat{\boldsymbol{\theta}}_i, \hat{\boldsymbol{\theta}}_{i-1})\right]\left[1-\alpha_2(\hat{\boldsymbol{\theta}}_i, \hat{\boldsymbol{\theta}}_{i-1}, \hat{\boldsymbol{\theta}}_{i-2})\right]\cdots}{\left[1-\alpha_1(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)\right]\left[1-\alpha_2(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \hat{\boldsymbol{\theta}}_2)\right]\cdots} \\ &\frac{\left[1-\alpha_{i-1}(\hat{\boldsymbol{\theta}}_i, \ldots, \hat{\boldsymbol{\theta}}_1)\right]}{\left[1-\alpha_{i-1}(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_{i-1})\right]}\Bigg\}\Bigg) \\ = \min\left(1, \frac{L_i}{G_i}\right).\end{aligned} \tag{13}$$

Also, we have that if the i-th stage is reached, it implies that $L_j < G_j$ for $j = 1, \ldots, i-1$, therefore, we replace $\alpha_j(\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots, \hat{\boldsymbol{\theta}}_j)$ with L_j/G_j and thereby obtain the recursive

formula

$$G_i = q_i(\hat{\boldsymbol{\theta}}_i|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots)[G_{i-1} - L_{i-1}], \tag{14}$$

which yields

$$\begin{aligned} G_i = q_i(\hat{\boldsymbol{\theta}}_i|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots)\big[q_{i-1}(\hat{\boldsymbol{\theta}}_{i-1}|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots)\big[q_{i-2}(\hat{\boldsymbol{\theta}}_{i-2}|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1, \ldots)\cdots \\ \big[q_2(\hat{\boldsymbol{\theta}}_2|\boldsymbol{\theta}_n, \hat{\boldsymbol{\theta}}_1)\big[q_1(\hat{\boldsymbol{\theta}}_1|\boldsymbol{\theta}_n)\pi(\boldsymbol{\theta}_n) - L_1\big] - L_2\big] - \cdots - L_{i-2}\big] - L_{i-1}\big]. \end{aligned} \tag{15}$$

The process of delaying rejection can be interrupted at any stage, and one way to achieve that is that upon each rejection, we toss a fair coin, if the outcome is head, we move to a higher stage proposal; otherwise, we let the chain stay where it is. The success of the DR approach basically depends on the fact that at least one of the proposals is chosen in the delaying process. Having separately discussed the AM and the DR methods, one possible combination strategy of the two approaches is hence described.

- Just as in AM, adapt the proposal at the first stage of DR. That is, using all previous points on the sampled chain, compute the first stage Gaussian proposal's covariance matrix, $\mathbf{C}_n^1$ via the AM recursion formula.

- For higher stages, say i-th stage $(i = 2, \ldots, m)$, compute the proposal covariance $\mathbf{C}_n^i$ as a scaled version of the proposal covariance of the first stage; $\mathbf{C}_n^i = \lambda_i \mathbf{C}_n^1$ with λ_i as a scaling factor which can be choosing freely.

Combining these two methods has clear benefits. The AM helps to adapt the first stage proposal to better fit the underlying target distribution. So in a case where the variance is too small or large, the points obtained from higher stages will help to properly transform the variance. Again, we have that sometimes, the initial guess for the proposal distribution is far from the actual one, thereby making it difficult to start the adaptation process. This usually occurs if the variance of the proposal is too large, or the proposal's covariance is almost singular. DR helps this situation by scaling down the size of the proposals at its higher stages, ensuring that some points will be accepted and with that, adaptation usually starts working properly.

2.3 Convergence Diagnostics of MCMC

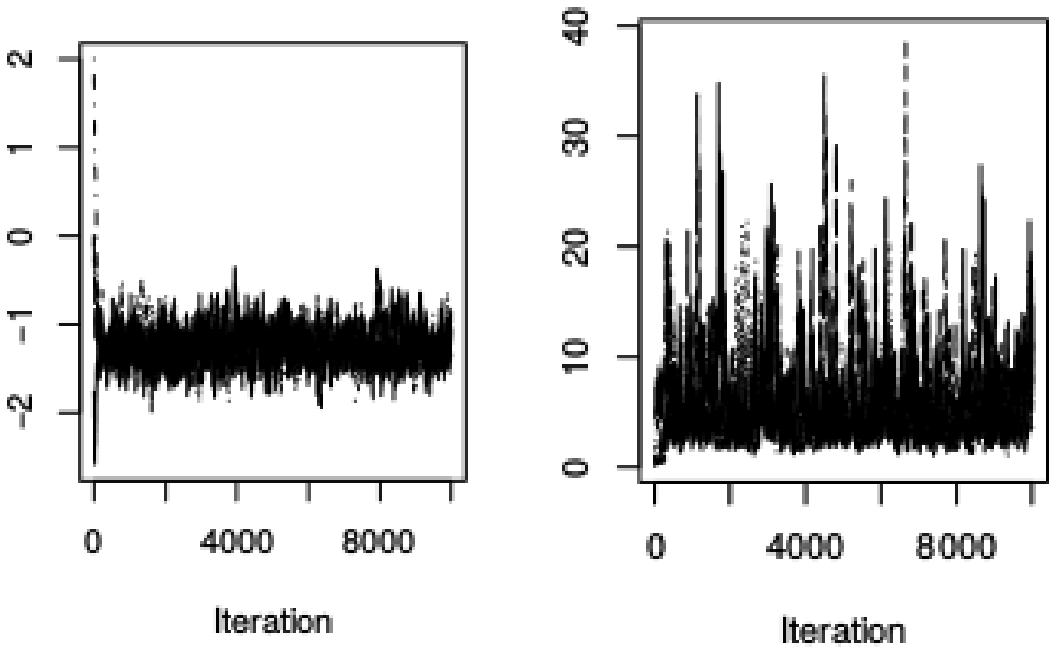
A major concern in using MCMC methods is determining when it is reasonable to stop sampling, with the believe that the samples are true representative of the underlying stationary distribution, so as to use the generated samples to estimate the characteristics of the distribution of interest. There are several diagnostics that can be done, both visual and statistical, to check if the chain appears to be converged, as discussed below.

2.3.1 Visual Inspection

One of the easiest way to check if the chain has converged is to visualise how well our chain is mixing or moving around the parameter space. This inspection should be done for every parameter. Thus, if the chain is taking a long time to cover the parameter space, then it will take longer to converge [32]. A number of graphical tools exist which can be easily implemented to provide useful feedback about the convergence of MCMC. We briefly describe these visual tools below.

- ***Time Series Plot.***

  According to [32], visualising the state of the chain through time is the most popular check for convergence of an MCMC algorithm. In this context, a time series plot is a plot that shows the chains of generated parameter values at each iteration in the algorithm. A line usually connects its successive points to allow for easy visualization of the path cruised by the chain. If different segments of a time series plot for a given parameter is shown to have traversed different parts of the sample space or if there is an obvious pattern, it implies that the MCMC algorithm, depicted by such a plot, may not have converged. In a case of multiple chains contained in the MCMC algorithm, a time series plot is made for the generated value of each parameter on a common graph. An evidence of non convergence in this case, is exhibited if the chains do not traverse the sample space in the same way. A visual illustration of time series plot is given in Figure 2.

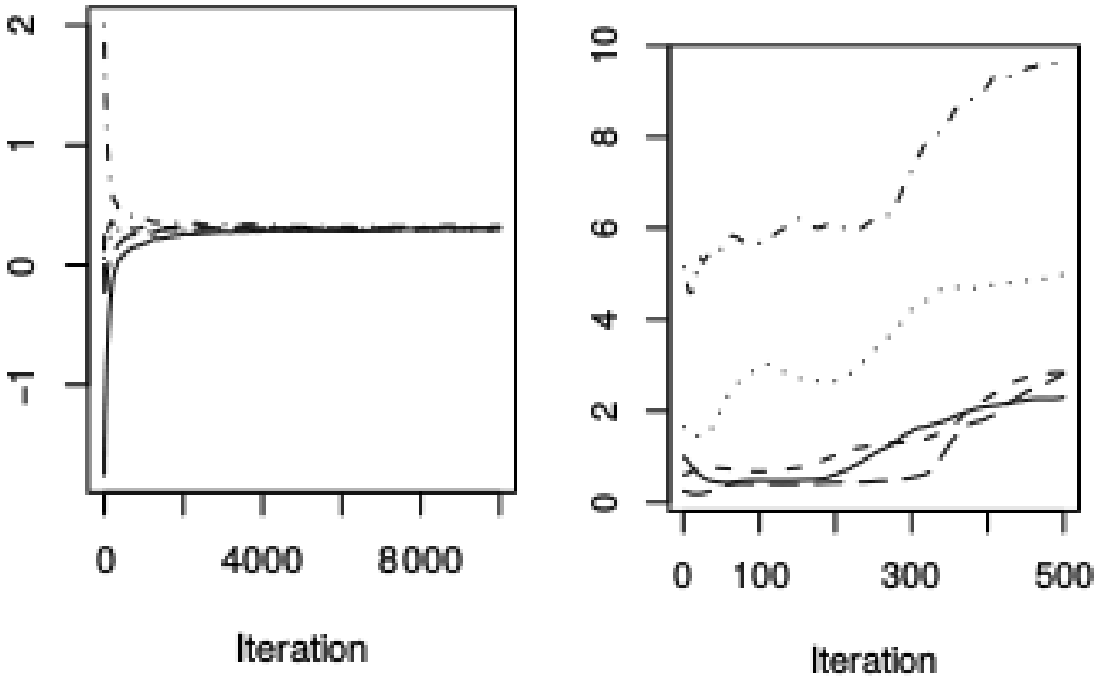


(a) (b)

Figure 2. Time series plot showing evidence of convergence (left) and lack of convergence (right) (Source:[32]).The plot gives the generated values of the parameter, in the y-axis against the iteration number. (a) Good mixing; (b) Bad mixing.

- ***Running Mean Plot.***

  The running mean plot can also be used to check how well the chains are mixing. According to [33], the running mean is the mean of all sampled values up to a given iteration. It is usually plotted after every n-th iteration and it is made as a plot of the iterations against the mean draws up to each iteration. With this kind of plot, an evidence of the convergence of the algorithm is noticed if the running mean stabilizes at the posterior mean for each parameter. A visual illustration of running mean plot is given in Figure 3 below.



(a) (b)

Figure 3. Running mean plot showing evidence of convergence (left) and lack of convergence (right) (Source:[32]). The plot gives the means of the parameter values sampled so far, in the y-axis against the iteration number. (a) Good mixing; (b) Bad mixing.

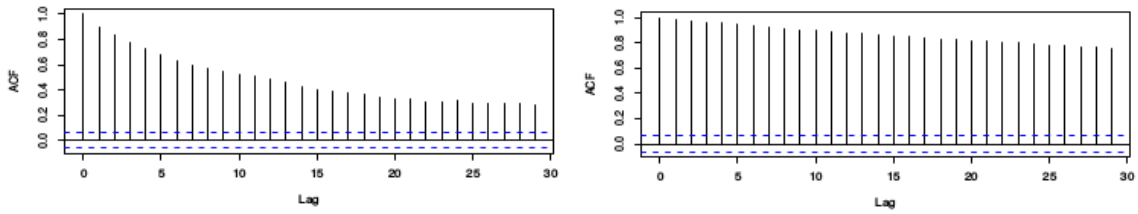
- ***Plot of Autocorrelation Functions ((ACF).***

  Another good practice in visualising how well the chain is mixing, is to make a plot of the autocorrelation functions of the parameter chain. It can be observed from the plot to what extent the samples that are k steps apart correlate with each other. The k-th autocorrelation ρ_k is given as

$$\rho_k = \frac{\sum_{i=1}^{n-k}(\boldsymbol{\theta}_i - \bar{\boldsymbol{\theta}})(\boldsymbol{\theta}_{i+k} - \bar{\boldsymbol{\theta}})}{\sum_{i=1}^{n}(\boldsymbol{\theta}_i - \bar{\boldsymbol{\theta}})^2}. \tag{16}$$

  Since in MCMC the next points depends on the previous points, we would expect subsequent points to correlate more with each other than points which are farther away. Thus, the k-th lag autocorrelation is expected to be smaller as k increases and as such, the 3rd and 4th draws should be more correlated than the 5th and 10th draws. If for higher values of k, autocorrelation is still high, it implies high degree of correlation between the draws and a slow mixing (chains traversing the sample

space slowly). Thus, an MCMC algorithm that generates highly autocorrelated parameter values will require a large number of iterations in order to transverse the whole sample space of the parameters [32]. An illustration of the ACF plot is shown in Figure 4.



(a) (b)

Figure 4. ACF plot showing evidence of convergence (left) and lack of convergence (right) (Source:[32]). (a) Good mixing; (b) Bad mixing.

2.3.2 Statistical Diagnostics

There are several statistical diagnostics that can be made in order to check for MCMC convergence. These include Gelman and Rubin, Geweke, Raftery and Lewis Convergence Measure. In this work, we describe the Gelman and Rubin Convergence Measure.

- ***Gelman and Rubin Convergence Measure.***

  According to [34] the Gelman and Rubbins [35] convergence diagnostics approach is currently the most popular diagnostics for convergence in the Statistics. Gelman-Rubin suggested running multiple chains with over dispersed starting values to compute the estimates of the posterior distribution and a reduction factor which shows how much sharper the estimate of the distribution might become if sampling were continued indefinitely. This approach of monitoring convergence is based on detecting when the chains are no longer influenced by their starting values, as large number of iterations are taken. The following steps are involved in this method:

  – Obtain an over dispersed estimate of the target distribution.

  – Generate starting values for the desired number of independent chain from the over dispersed estimate of the target distribution.

– Compute the within-chain, W and between-chain, B variance as;

$$W = \frac{1}{m}\sum_{j=1}^{m} s_j^2, \qquad \text{where } s_j^2 = \frac{1}{n-1}\sum_{i=1}^{n}(\boldsymbol{\theta}_{ij} - \bar{\boldsymbol{\theta}}_j)^2 \tag{17}$$

$$B = \frac{n}{m-1}\sum_{j=1}^{m}(\bar{\boldsymbol{\theta}}_j - \bar{\bar{\boldsymbol{\theta}}})^2 \qquad \text{where } \bar{\bar{\boldsymbol{\theta}}} = \frac{1}{m}\sum_{j=1}^{m}\bar{\boldsymbol{\theta}}_j \tag{18}$$

– Calculate the estimate of the variance of the stationary distribution as a weighted sum of the within-chain and between-chain variance as;

$$\hat{Var}(\boldsymbol{\theta}) = \left(1 - \frac{1}{n}\right) W + \frac{1}{n} B, \tag{19}$$

which is unbiased if the starting distribution is equal to the stationary distribution.

– Compute the potential scale reduction factor as;

$$\hat{R} = \sqrt{\frac{\hat{Var}(\boldsymbol{\theta})}{W}} = \sqrt{\frac{n-1}{n} + \frac{1}{n}\frac{B}{W}}, \tag{20}$$

where n is the number of iterations which is sometimes taken after discarding the first few iterations. Since the initial value of the chains are over dispersed, we have that the quantity B is initially larger than W and as such, the potential scale reduction factor is considerably larger than one initially. As large numbers of iteration are taken, the potential scale reduction factor declines to 1 if the algorithm converges. At this point, the chains are no longer impacted by their starting values and have cruised the whole sample space.

Many other methods for the MCMC convergence diagnostics exist [32], but in this work, we discussed the commonly used approaches.

In this study, we employ the basic features of MCMC in modelling the host-seeking behavior of the target mosquito species (*An. gambiae* and *An. arabiensis*) as well as for the identification of model parameters.

3 MODEL FORMULATION AND SIMULATION CONFIGURATION

The modeling approach alongside the selections and parametrization of the factors required for the modeling is presented in this section. The assumptions of the models are also stated herein.

3.1 Modeling Mosquito Movement and Attraction to Host

3.1.1 Initialising Positions and Conditions

Initially in the simulations, the mosquitoes are represented as a number of agents in a rectangular patch $[\mathrm{x}_{\min}, \mathrm{x}_{\max}] \times [\mathrm{y}_{\min}, \mathrm{y}_{\max}]$ at uniformly random spatial positions generated with the formula

$$\begin{aligned} \mathbf{x}_{\text{cord}} &= \mathrm{x}_{\min} + (\mathrm{x}_{\max} - \mathrm{x}_{\min}) \times \mathbf{r}, \\ \mathbf{y}_{\text{cord}} &= \mathrm{y}_{\min} + (\mathrm{y}_{\max} - \mathrm{y}_{\min}) \times \mathbf{r}, \end{aligned} \tag{21}$$

where $\mathbf{r} \sim U(0, 1)$, is a vector of random numbers from the uniform distribution with length, corresponding to the number of mosquitoes N_m. Thus, we have an $(\mathrm{N}_m \times 2)$ matrix of initial positions of the mosquitoes. The initial distance of each mosquito from the human is thereby calculated as

$$\mathbf{d}_{\text{old}} = \sqrt{\mathbf{x}_{\text{cord}}^2 + \mathbf{y}_{\text{cord}}^2}, \tag{22}$$

and the initial concentration of attractive odour with respect to the positions is equally calculated as discussed in details at the later part of this subsection. Furthermore, the property list of each mosquito agent which is updated at each timestep of the simulation, are given initially as a matrix of size $\mathrm{N}_m \times 12$, where such properties are 12 in number as defined below.

Table 1. Property lists of mosquitoes.

Conditions	Properties	Definition
1	In	Equation 23
2	Trapped (exited)	Equation 40
3	Dead	Equation 59
4	Fed	Equation 61
5	Time indoors	Equation 62
6	Klinotaxis	Equation 64
7	Inside net	Equation 38
8	Time spent in one position	
9	Concentration of chemical	Equation 53
10	Hitted net	Equation 65
11	Hitted wall	Equation 67
12	Time after hitting the net	Equation 65.

The detailed explanation of these conditions are given later in this section as precisely specified in the above table. First, we check if the mosquitoe's position enter the rectangular patch (the hut), using condition 1, as we check if its old distance is less than or equal to the hut size, denoted as h_s, as given in the equation below

$$\text{condition}(:, 1) = (\mathbf{d}_{\text{old}} \leq h_s). \tag{23}$$

Generally, in the hut experiment we place all the mosquitoes inside of the hut at the beginning of the simulation. So, this condition(:,1) is redundant, and is more used in the community level experiment. Thus, given that all mosquitoes at the hut-level experiment have initially gained entrance into the hut, we model their candidate steps and attraction to the host as described below.

3.1.2 Attraction Model

The mosquito attraction model is based on the assumption that a mosquito estimates the direction of odour increase (the gradient) from the host by the mechanism of klinotaxis. During this plume-tracking behavior, a mosquito samples the host odour at one location, then changes location and repeats the sampling, as it uses its memory of the concentrations previously encountered to choose the next position [13].

Imitating the aforementioned mechanism, we can model the flight of mosquitoes as a discrete time-stepping correlated random walk. Suppose that a mosquito agent is at position $\mathbf{x}^{n-1}$ at time step $n-1$, then the mosquito can randomly select a new position $\mathbf{x}^n$; which relates to the first position by the formula

$$\mathbf{x}^n = \mathbf{x}^{n-1} + \Delta\mathbf{W}, \tag{24}$$

from a two-dimensional proposal distribution. The increment, $\Delta\mathbf{W} \sim \mathbf{N}(R, \Sigma)$ is a random point on a circle centered at the origin of the coordinate system (0,0). Thus the next position, $\mathbf{x}^n$ is sampled as a random point on a circle centered at $\mathbf{x}^{n-1}$ with radius; $R = 0.4$m, with random numbers generated from $\mathbf{N}(0, \sigma^2)$ and having $\sigma = 0.1$ added in the direction of the radius. In this work, $\Sigma = \sigma^2 I$ where I is an identity matrix and the parameters $\mathbf{x}_0$ and σ were matched with the intent that they imitate the real speed of mosquito flight which lies in the range $0.4 - 1.1$ m/s, for most mosquito species [36]. Thus we have that the increment with respect to the coordinates is given in the equation below

$$\Delta\mathbf{W} = \mathbf{d}_i \left[\cos(\phi) - \sin(\phi)\right] \tag{25}$$

where $\mathbf{d}_i$ is the displacement of the n-th movement and it is calculated using the formula

$$\mathbf{d}_i = 0.4 + 0.1 \cdot \mathbf{r}, \tag{26}$$

where $\mathbf{r} \sim N(0, 1)$ with length, corresponding to the number of mosquitoes. The sampling of the increment is done in a similar way at every iteration of the algorithm. Thus the random direction (in a circle) taken by each mosquito is given as

$$\phi = 2\pi \cdot \mathbf{r}, \tag{27}$$

where $\mathbf{r} \sim U(0, 1)$ with length, corresponding to the number of mosquitoes. The mosquito flight is given by the above random walk in the absence of attraction effects towards the host.

However, when there are attraction effects, we shall employ the main features of Metropolis algorithm in order to simulate the movement of mosquito towards the host. As discussed in Section 1.1.3, that attractive odours emitted by the host are helpful clues for mosquitoes to locate the hosts. We shall model the concentration of these attractive odours and the area covered by the odour using the solution of the diffusion equation, with only the diffusive spread of the odour taken into account. Thus, a particular position where the host is situated can be said to be the region of high concentration of odour and the maximal distance at which the mosquito is able to sense the host is viewed as the region of low

concentration. This aligns with the concept of the diffusion equation which describes the expel of the flow of some quantity (intensity, temperature) over space [37]. Thus for the hut experiment, we use the solution of the diffusion equation given as

$$C(\mathbf{x}, \mathbf{x}^h) = \exp\left[-\frac{\|\mathbf{x} - \mathbf{x}^h\|^2}{2\sigma_a^2}\right], \tag{28}$$

with point source, the Gaussian Kernel centered at a spatial position of the host. The concentration $C(\mathbf{x}, \mathbf{x}^h)$ at any point, obeys the Gaussian bell shape [38]. In Equation 28, $\mathbf{x}$ denotes the mosquito position, $\|\mathbf{x}-\mathbf{x}^h\|$ is the distance from which the mosquito senses the host with the influence of odour concentration, $C(\mathbf{x}, \mathbf{x}^h)$ and the standard deviation of the Gaussian, σ_a determines the maximal distance at which the mosquito is able to sense the host.

In a community level simulation, where there are multiple hosts, the total concentration of attractive odour at point $\mathbf{x}$, can be measured as the sum of all the individual concentrations measured at $\mathbf{x}$, corresponding to individual hosts, and it is given as

$$C_a(\mathbf{x}) = \sum_{i=1}^{N_h} \left\{ \exp\left[-\frac{\|\mathbf{x} - \mathbf{x}_i^h\|^2}{2\sigma_a^2}\right] \right\}, \tag{29}$$

where $\|\mathbf{x} - \mathbf{x}_i^h\|$ is the distance between the position of mosquito $(\mathbf{x})$ and position of i^{th} host, $\mathbf{x}_i^h$ and N_h is the total number of human hosts.

Thus, the attraction is given by means of an accept-reject procedure as described by the Metropolis algorithm. Suppose that we take the previous and new position of a mosquito to be $\mathbf{x}^{n-1}$ and $\mathbf{x}^n$ respectively, with respective probabilities as p_{n-1} and p_n, we have that the mosquito accepts the new point with probability

$$\alpha_a(\mathbf{x}^n|\mathbf{x}^{n-1}) = \min\left(1, \frac{p_n}{p_{n-1}}\right), \tag{30}$$

where p_n/p_{n-1} is the ratio of the attraction potential function $p(\mathbf{x})$ defined at each point $\mathbf{x}$, which depends on the concentration and other attraction factors. So we have that steps upwind $(p_n > p_{n-1})$ are always accepted while steps downwind $(p_n < p_{n-1})$ may as well be accepted with probability, p_n/p_{n-1}.

Among the attraction factors, CO_2 is regarded as the main attraction factor for the mosquitoes [39]. In order to parsimoniously define the other short-distance attraction factors, we de-

fine the attraction potential function as

$$p(\mathbf{x}) = \exp\left(\frac{C(\mathbf{x})}{\sigma_{acc}}\right), \tag{31}$$

where σ_{acc} is a scaling factor that depends on the distance to the host.

The above equation can further be modified to be

$$p(\mathbf{x}) = \exp\left(\frac{C(\mathbf{x}^n, \mathbf{x}^h) - C(\mathbf{x}^{n-1}, \mathbf{x}^h)}{\sigma_{acc}}\right), \tag{32}$$

where $C(\mathbf{x}^{n-1}, \mathbf{x}^h)$ is the initial concentration of attractive odour and $C(\mathbf{x}^n, \mathbf{x}^h)$ is the new concentration of attractive odour. Intuitively, if the new concentration, is greater than the old concentration, the step is always accepted, otherwise, the step is tentatively accepted based on some criteria as discussed in the last section. The above acceptance probability includes all the influence of treated and untreated nets and also the hut barrier which is given by a rejection function.

Thus, we update mosquito position if they have actually moved and if the candidate step is accepted as given below:

$$\text{update position} = \text{move } \ \& \ \text{attr } \ \& \ \text{acc}, \tag{33}$$

of which a mosquito can move if it is not dead and not trapped, defined as

$$\text{move} = \sim \text{condition}(:, 2) \ \ \& \sim \text{condition}(:, 3). \tag{34}$$

The mosquitoes that actually moved, achieved this, based on the criteria given by 'attr' which is different for treatment and control case as defined below

$$\text{attr} = \begin{cases} u_{acc} < \alpha_a(\mathbf{x}^n|\mathbf{x}^{n-1}) \ \mid \ \text{condition}(:, 6), & \text{control case} \\ u_{acc} < \alpha_a(\mathbf{x}^n|\mathbf{x}^{n-1}) \cdot (1 - \alpha_r(\mathbf{x}|d_{50}, r, s)) & \\ \quad \mid \ u_{acc} < (1 - \alpha_r(\mathbf{x}|d_{50}, r, s)) \ \ \& \ \ \text{condition}(:, 6), & \text{treatment case}, \end{cases} \tag{35}$$

where u_{acc} is the set of random numbers generated for the accept/reject procedure for candidate steps and $\alpha_r(\mathbf{x}|d_{50}, r, s)$ is the probability of repulsion by a repellent which is discussed later in details. Finally, the third criteria for updating a mosquito step, is that which depicts the acceptance of a candidate step made and described as

$$\text{acc} = \sim \text{exited hut } \ \& \ \ \sim \text{entered net } \ \& \ \ \sim \text{unphy jump}, \tag{36}$$

where ‘entered net’ is used to describe the state of a mosquito whose proposed step falls inside the net and is primarily accepted, whereas the old position is outside and the mosquito was not able to gain full entrance into the net (accept/reject procedure) thereby, accounting for net barrier. In this situation, we simulate the mosquito hitting the net (see Section 3.4). The above description is defined as

$$\text{entered net} = \sim \text{condition}(:, 7) \ \ \& \ \ (\mathbf{d}_{\text{new}} < d_p). \tag{37}$$

Hence, we note that ‘entered net’ is different from a mosquito being inside the net which is given by condition 7 and defined as

$$\text{condition}(:, 7) = \mathbf{d}_{\text{old}} < d_p. \tag{38}$$

In continuation of the definition of Equation (36), we have that, ‘exited hut’ is defined as

$$\text{exited hut} = \text{condition}(:, 1) \ \ \& \ \ (\mathbf{d}_{\text{new}} > h_s) \ \ \& \ \ \sim \text{condition}(:, 2). \tag{39}$$

Thus, this has a similar definition with that of ‘entered net’ though logically, the difference is that, it is used to account for the barrier created by the wall and we simulate the mosquito hitting the wall (see Section 3.4) when this happens. Again, we note that ‘exited hut’ is not the same as the mosquito actually exiting the hut as given by condition 2 and defined as

$$\text{condition}(:, 2) = \text{condition}(:, 1) \ \ \& \ \ (\mathbf{d}_{\text{old}} > h_s), \tag{40}$$

and the mosquito is marked as trapped under such condition.

Next we define the unphysical jump through the net, made by mosquitoes as

$$\begin{aligned} \text{unphys jump} = (\mathbf{d}_{\text{old}} \geq d_p) \ \ \& \ \ (\mathbf{d}_{\text{new}} \geq d_p) \ \ \& \ \ \text{Discr} \geq 0 \\ \& \ \ (t_1 \geq 0) \ \ \& \ \ (t_1 \leq 1) \ \mid \ (t_2 \geq 0) \ \ \& \ \ (t_2 \leq 1) \ , \end{aligned} \tag{41}$$

where the discriminant factor is defined as

$$\text{Discr} = b^2 - 4ac, \tag{42}$$

and the quadratic coefficients are derived by finding the intersection of the line and a circle

(gradient) and based on natural parameterization, we have

$$\begin{aligned} a &= \mathbf{x}_{\text{diff}} + \mathbf{y}_{\text{diff}}, \\ b &= 2(\mathbf{x}_{\text{prod}} + \mathbf{y}_{\text{prod}}) - (\mathbf{x}_{\text{cord}} + \mathbf{y}_{\text{cord}}), \\ c &= (\mathbf{x}_{\text{cord}} + \mathbf{y}_{\text{cord}}) - d_p^2, \end{aligned} \tag{43}$$

given that $\mathbf{x}_{\text{diff}}$ and $\mathbf{y}_{\text{diff}}$ give the difference between the old and new coordinates and also, $\mathbf{x}_{\text{prod}}$ and $\mathbf{y}_{\text{prod}}$ denote an element-wise product of the old and new coordinates. Again, note that $\mathbf{x}_{\text{cord}}$ and $\mathbf{y}_{\text{cord}}$ are the coordinates that constitute the old position of a mosquito. Hence, we have that t_1 and t_2 are the resulting roots of the quadratic equation defined as

$$\begin{aligned} t_1 &= \frac{-b + \sqrt{\text{Discr}}}{2a}, \\ t_2 &= \frac{-b - \sqrt{\text{Discr}}}{2a}. \end{aligned} \tag{44}$$

Based on these descriptions, if the criteria stipulated in Equation 33 is met by any mosquito at any time-step in the simulation, the coordinates alongside the distance from the host and the corresponding concentration of attractive odor is updated for that mosquito.

Furthermore, an intuitive annotation of σ_{acc} in Equation 32 stems from the fact that, we assume that at a short distance to the host, the ability of the mosquito to sense is enhanced, since it becomes able to identify the prey via vision and heat sensors [40]. In order to account for the increasing greediness of the mosquito in the odour plume as a result of vision activated at short distance to the host, we introduce a linear distance dependency of the scaling factor as

$$\sigma_{acc}(\mathbf{x}) = \begin{cases} \sigma'_{acc} + \sigma''_{acc}\|\mathbf{x} - \mathbf{x}^h\|, & \text{for } \|\mathbf{x} - \mathbf{x}^h\| \leq 80 \\ \sigma_{acc}^{\max}, & \text{for } \|\mathbf{x} - \mathbf{x}^h\| > 80 \end{cases} . \tag{45}$$

The above function increases from σ'_{acc}; which is the minimum value (i.e the value of the scaling factor when the distance to the host is zero), with a slope given by σ''_{acc} until it is replaced by a constant, which suitably provides a purely random movement outside the concentration plume (see Section 3.6).

For the community scale simulation, it is assumed that the scaling factor σ_{acc} depends on the distance to the closest host. The algorithm that implements this, imitates that of the Simulated Annealing optimization method [41], with the 'annealing temperature schedule' replaced with the 'greediness scale'. Figure 5 gives an overview of the scaling

factor, σ_{acc} as well as the resulting probability of accepting steps away from the host.

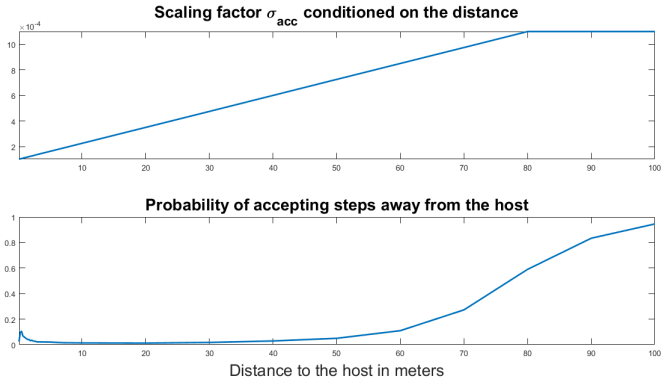


Figure 5. Scaling factor σ_{acc} conditioned on distance to the host (top), average probability of accepting candidate steps taken away from the host; as a function of distance to the host (bottom). (Source:[23])

As shown in Figure 5, the probability of accepting steps away from the host attenuates as the mosquito approaches the host. However, this probability locally increases, just a bit as the mosquito gets very close to the host. This is a typical property exhibited by mosquitoes, as explained in [42] where *Anopheles gambiae* was observed to exhibit more tortuous flights as it scans the environment before landing.

The table below presents the formulas of the models formed in this subsection and the names of the functions/scripts for which they are executed in MATLAB.

Table 2. Trace back to MATLAB.

Model executions	MATLAB script/function names
Equations (21) and (22)	init_positions.m
Equation 23	mosq_entered.m
Equations (24), (25) and (27)	candidate_step.m
Equation 26 and random number generations	gen_randoms.m
Equations (28), (30) and (32)	hut_experiment.m
Equation 33	mosq_update_position.m
Equation 34	death.m
Equation 35a	control.m
Equation 35b	LLIN.m
Equations (36), (37), (39) and (41) to (44)	physical_barrier6.m
Equation 38	mosq_inside_net.m
Equation 40	mosq_exit.m
Equation 45	hut_experiment.m .

3.2 Modeling Protective Measures

The protective measures considered in this work; untreated and LLINs, are modelled using the properties of the logistic function given as

$$y = \frac{1}{1 + \exp(-x/s)}. \tag{46}$$

The logistic function is used to describe certain kinds of growth rate that has an S-shaped behaviour. This function grows exponentially at first, but due to certain restrictions, eventually grow more slowly and levels off as $|x| \to \infty$.

However, in modeling protective systems, we aim at getting the probability of rejection at a candidate mosquito position, $\mathbf{x}$. Thus, we should have a function that describes the fact that, repulsion effect amplifies as the mosquito approaches the source of repellent. In doing this, we modify the above logistic equation as

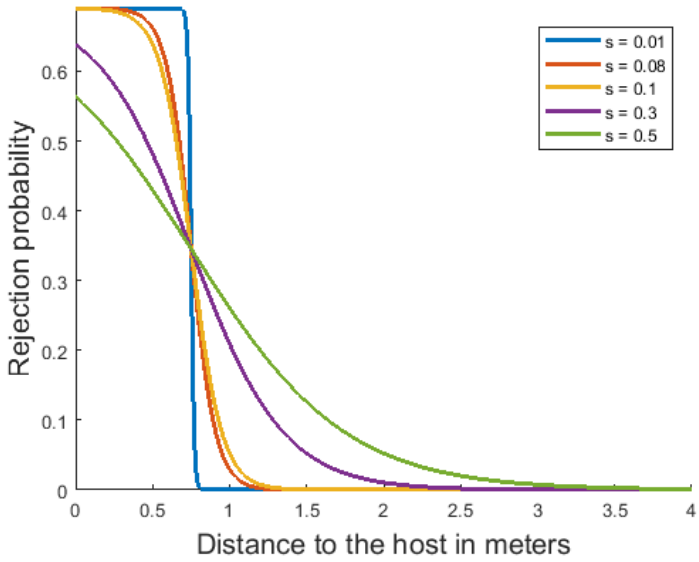
$$\tilde{y} = 1 - \left(\frac{1}{1 + \exp(-x/s)} \right). \tag{47}$$

Furthermore, in order to capture the protective properties of the nets, we shall modify

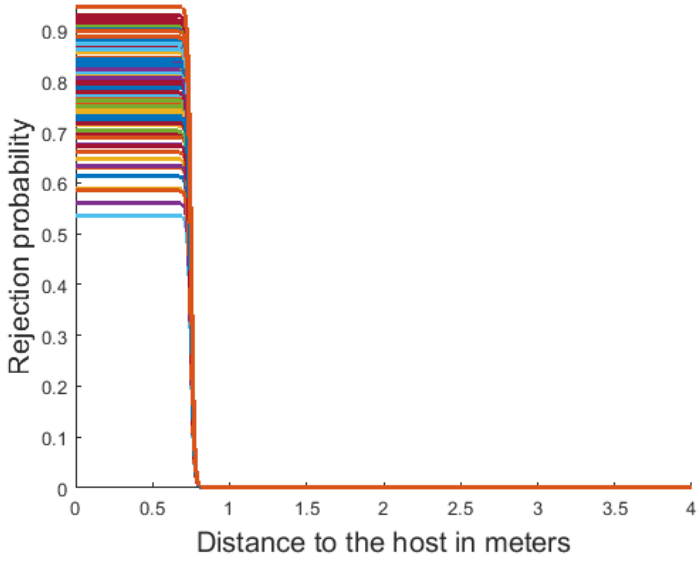
Equation 47 above to obtain a distant dependent function given as

$$\alpha_r(\mathbf{x}|d_{50}, r, s) = r\left[1 - \frac{1}{1 + \exp\big(-(\|\mathbf{x} - \mathbf{x}^h\| - d_{50})/s\big)}\right], \tag{48}$$

where: $\|\mathbf{x} - \mathbf{x}^h\|$ is the distance from the mosquito to the protected human, s determines the range of repellent that causes repulsion with non zero probability, r denotes the intensity of repulsion (since different repellent have different repulsion intensity), d_{50} stands for the distance from the host where the concentration of repellent attains 50% of its total and at this distance, the probability of repulsion has the value of 0.5. The above formula models the rejection probability at the candidate position, $\mathbf{x}$. Furthermore, Figure 6 generally illustrates how the probability of rejection changes with respect to distance to the host, the repulsive range and the repellent intensity. Thus, Figure 6a features the probability of rejection with a randomly sampled repellent intensity of 0.6905 from the MCMC chains generated for IconMaxx chemical and with different repulsive ranges, defined by s. However, since most of the treatment chemicals act at a short distance, s was fixed to 0.01 in the simulation. Thus, Figure 6b presents the the rejection probability curves for 100 randomly sampled repellent intensity values, r from the MCMC chains generated for IconMaxx chemical in a simulated hut experiment, with slope, $s = 0.01$.



(a)



(b)

Figure 6. Rejection probability for different values of s and r, with $d_{50} = 0.75$. (a) Probability of rejection with different values of s. (b) Probability of rejection with s fixed at 0.01.

The formula presented in Equation (48) is applied when the mosquito is still outside the net. However, if it gains entrance into the net, the direction of net repulsion is swapped such that the repulsion probability reduces as it approaches the host under the net. Therefore, the probability of repulsion is generally modeled as

$$\alpha_r(\mathbf{x}|d_{50},r,s)=\begin{cases} r\left[\frac{1}{1+\exp\left(-(\|\mathbf{x}-\mathbf{x}^h\|-d_{50})/s\right)}\right], & \text{for 'inside net'} \\ r\left[1-\frac{1}{1+\exp\left(-(\|\mathbf{x}^i-\mathbf{x}^h\|-d_{50})/s\right)}\right], & \text{for 'outside net'} \end{cases}. \tag{49}$$

3.2.1 Untreated Nets

Nets posing just a physical barrier to mosquitoes conform with rejection probability given as a step function which is a particular case of Equation 48 where $s << 1$. With this, the mosquito can roam around the net with virtually zero probability of getting inside. We consider the fact that realistically, some of the bed nets used in rural communities are commonly holed and as such, deliberately holed nets are commonly used in hut trials [43]. Hence, we simulate torn nets such that they have a non zero probability for mosquito penetration. Each of the six holes simulated has a size of $\pi(1-p_{net})$. Thus, we assign a penetration probability of $1-p_{net}<1$ for torn nets, where p_{net} denotes the probability of being blocked by a net barrier and so we have that for the simulations, the acceptance probability of a proposed mosquito step inside the net area is $1-p_{net}$.

3.2.2 Treated Nets

For the treated nets, we model the repulsion by the LLIN in two stages. We first start with applying the accept/reject step as given by the Metropolis type of probability

$$\alpha_r(\mathbf{x}^n|\mathbf{x}^{n-1})=\min\left(1,\alpha_r(\mathbf{x}|d_{50},r,s)\right), \tag{50}$$

where α_r stands for the probability of rejection. We further take into account, the physical barrier posed by nets just like that of the untreated net. However, the poisoning effect which the LLIN are equipped with, is such that, the probability of insecticide induced mortality of a mosquito increases upon consuming the chemical substance spread on the net surface.

The function which executes these repulsion effects given by Equations (48) to (50) is named prob_rep_vec.m in the MATLAB codes.

3.3 Modeling Mortality Rates

We model the death rates of mosquitoes with cognizance of the two kinds of protective measures considered in this work. In the control case, mosquitoes can only die a natural death and as reported in the literature [44], a 34 hour death rate, μ of 10% is typical for both *An. gambiae* and *An. arabiensis*. The death rate is possibly enhanced in the treatment case considering the fact that it takes the poisoning effect of the chemical substance sprayed on the net surface into account.

Thus, we model the total consumed dosage of chemical using the solution of the diffusion equation, with the Gaussian kernel centered at a spatial position of the net, as a point source. Thus, the dosage of poisonous chemical is defined as

$$\mathrm{D}(\mathbf{x}, \mathbf{x}^n) = \sum_{i=1}^{T} \exp\left[-\frac{\|\mathbf{x} - \mathbf{x}^N\|^2}{2\sigma_\mu^2}\right], \tag{51}$$

of which we assume that the dosage of the chemical is taken only when contacting with the net, not at a distance, hence, there is a unit dosage of chemical that can be computed by replacing the norm with actual size of the net, d_p. The standard deviation of the Gaussian, σ_μ denotes the effective range of the poison. Thus, the total accumulated dosage of the chemical is dependent on the sequence of positions; $\mathrm{D}(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n, \mathbf{x}_N)$. Hence, a mosquito is able to accumulate chemical based on the criteria described in the equation below

$$\text{take chemical} = \text{move } \ \& \ \text{ entered net}, \tag{52}$$

where the above two criteria are defined in Equation (34) and Equation (37) respectively. Thus, the dosage of poison (given in condition 9) for such mosquito which satisfy Equation (52) is updated as

$$\text{condition(take chemical,9)} = \text{condition(take chemical,9)} \ + \mathrm{D}(\text{take chemial}, 9), \tag{53}$$

where D denotes the total sum of dosages calculated over all the contacts with the net. Hence we account for insecticide induced mortality rate using the properties of logistic equation with slight modification such that it captures the poisoning properties of the net. This mortality rate is dependent on the effective amount of chemical consumed by a mosquito up to a particular instant in time t and so we have a dosage dependent function parameterized as

$$\mu_c = \frac{t_\mu}{1 + \exp\big(-(\mu_d - d_{50}^\mu)/s_\mu\big)}, \tag{54}$$

where t_μ is the chemical toxicity which is different for different insecticides, d_{50}^μ is the

critical dosage of chemical at which death happens with 0.5 probability, s_μ is the temporal range of poison and it determines the range of consumed dosage that cause death with non-zero probability and μ_d is the consumed dosage of chemical as given in Equation (53). Figure 7 illustrates the increase in the chemically induced death rate based on the amount of consumed dosage.

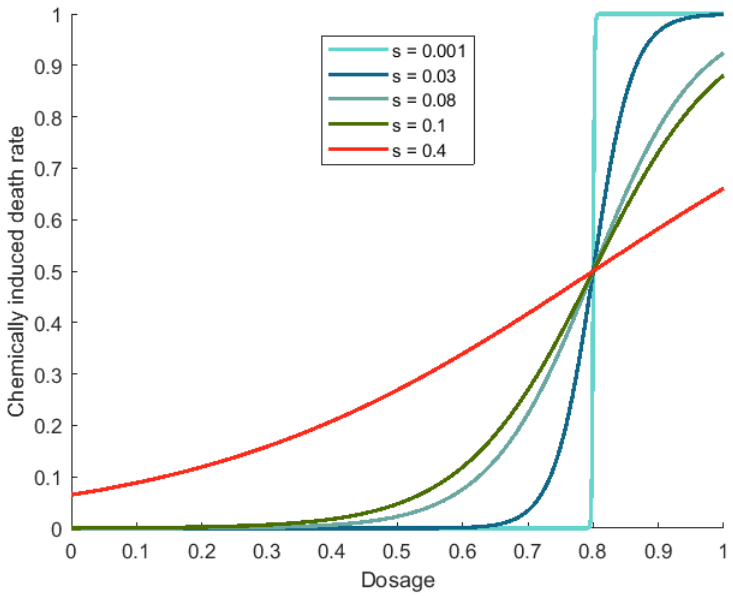


Figure 7. Insecticide induced death rates for different values of s_μ, with $d_{50}^{\mu} = 0.8$ and $t_\mu = 1$.

In implementation s_μ was taken to be 0.001 which gives a step function such that the the chemical induced death rate for any mosquito becomes 1 if the consumed dosage is greater or equal to d_{50}^{μ} and zero, otherwise.

However, in continuous time, we usually express the natural mortality of a population as the ordinary differential equation given in Equation 55

$$\frac{dP}{dt} = -\mu P, \tag{55}$$

where μ and P denote the decay rate and the population size respectively. Adopting the above formula to capture the chemically enhanced death, we can re-write it as

$$\frac{dP}{dt} = -(\mu + \mu_c)P, \tag{56}$$

with μ_c, being the insecticide induced decay rate. Relating it with the discrete-time cal-

culations, we discretize the above equation as

$$\frac{P(t+\Delta t)-P(t)}{\Delta t} = -(\mu+\mu_c)P(t), \tag{57}$$

where $\Delta t = t_n - t_{n-1}$ is a time unit. With some mathematical manipulations below;

$$\begin{aligned}
\frac{P(t+\Delta t)}{\Delta t} &= \frac{P(t)}{\Delta t} - (\mu+\mu_c)P(t),\\
\frac{P(t+\Delta t)}{\Delta t} &= \frac{P(t)(1-(\mu+\mu_c)\Delta t)}{\Delta t},\\
P(t+\Delta t) &= P(t)(1-(\mu+\mu_c)\Delta t),
\end{aligned}$$

we have that the above equation becomes

$$\frac{P(t+\Delta t)}{P(t)} = 1-(\mu+\mu_c)\Delta t, \tag{58}$$

which describes the relative decrease in the size of the population at a given time step. From the above equation, the total probability of death per unit change in time, Δt is parametrized as

$$\alpha^{\Delta t} = \min\{1, (\mu+\mu_c)\Delta t\}, \tag{59}$$

where $\Delta t = 2$ seconds, is the length of time interval. Therefore at each step in the simulations, the total probability of death for every mosquito agent is computed as the sum of both the chemically induced and natural death rates. Hence we have that in the control case, $\mu_c = 0$ and at the initial stage in the treatment case, when the mosquito is free from chemical poison, the death rate is restricted to be a natural mortality but after it accumulates a lethal dose of chemical, its total probability of death, $\alpha^{\Delta t}$ tends to one. The probability of death is tracked and updated separately for each mosquito in the simulation.

Furthermore, we mimic the process of accounting for delayed mortality as demonstrated by [22], where mosquitoes that have already stayed for 10 hours in the hut, were collected alive and kept for 24 hours under the glass, before scoring for delayed mortality. In doing this, we fix the death rates after 10 hours in the hut and account for probability of death after a time period of 24-hours as

$$\alpha_{24\text{h}}^{\Delta t} = \min\left\{1, \left[\mu_{10h}^{\Delta t} + \mu_c\right]\Delta t\right\}, \tag{60}$$

where $\mu_{10h}^{\Delta t}$ is the 10-hour natural death and $\Delta t = 24 \cdot 1800$. Hence we have that a mosquito is marked as dead if the total probability of death after 34 hours is less than some generated random number from the uniform distribution. If a mosquito is marked as dead,

the property list and position of such mosquito will no longer be updated. However, it is necessary to note that the mortality status of a trapped mosquito as given in Equation (40) is the only property of such mosquito that is updated at every step of the simulation.

In continuation of the description of the criteria for updating the property list of mosquitoes, we have that a mosquito is scored as fed if its updated position is closer to the center point of the host than ϵ (see Table 5) as given in condition 4.

$$\text{condition}(:,4) = \sim \text{condition}(:,3) \ \ \& \ \ \mathbf{d}_{\text{old}} < \epsilon. \tag{61}$$

Hence, if marked as fed or if the maximal time t_{max} that it should spend for host-seeking is used up, the mosquito switches to a pure random walk. The indoor time for each mosquito, as given in condition 5 is incremented with the formula

$$\text{condition}(\text{inside alive}, 5) = \text{condition}(\text{inside alive}, 5) + 2, \tag{62}$$

where 'inside alive' is a logical vector, with respect to each mosquito and defined as

$$\text{inside alive} = \text{condition(:,1)} \ \ \& \sim \text{condition(:,2)} \ \ \& \sim \text{condition(:,3)}. \tag{63}$$

Intuitively, the time spent indoors for all mosquitoes that are inside the hut and are neither dead nor trapped are updated by 2 seconds. To that end, we have that a mosquito switches to a pure random walk as given by condition 6, based on the formula

$$\text{condition}(:,6) = \text{condition}(:,4) \ \ | \ \ (\text{condition}(:,5) \geq t_{max}) \tag{64}$$

and it should be noted that the barriers posed by the net and the repellent effect alongside the impact of chemical poisoning still remain functional under this condition.

The table below presents the formulas of the models formed in this subsection and the names of the functions/scripts for which they are executed in MATLAB.

Table 3. Trace back to MATLAB.

Model executions	MATLAB script/function names
Equation 51	conc_pois.m
Equations (52) and (53)	mosq_acum_chem.m
Equation 54	prop_enh_death.m
Equation 59	death.m
Equation 60	statistics.m
Equation 61	mosq_fed.m
Equations (62) and (63)	increment_indoor_time.m
Equation 64	switch_to_kinesis.m .

3.4 Accounting for Net and Hut Barriers

We further explicitly explain how to treat the movement at the net, inside the net, on the walls of the hut and for exiting the hut. If a candidate step gains coordinates inside the net and it is rejected with probability, p_{net}, we update the mosquitoes position with the nearest point on the net (lying in between the old position and the new position) than that of position $\mathbf{x}^{n-1}$ such that $\parallel \mathbf{x} - \mathbf{x}^h \parallel = d_p$, where d_p is the net width as shown in Table 5. Thus, we mark the mosquito hitting the net as well as the time it spends after hitting the net as

$$\begin{aligned} &\text{condition}(\text{ind\_hit } \& \ \sim \text{acc\_net}, 10) = 1, \\ &\text{condition}(\text{ind\_hit } \& \ \sim \text{acc\_net}, 12) = 2, \end{aligned} \tag{65}$$

where ind_hit and acc_net are modeled as

$$\begin{aligned} &\text{ind\_hit} = \text{move } \& \text{ attr } \& \text{ (entered net } \mid \text{ unphys junp)}, \\ &\text{acc\_net} = u < 1 - p_{net}, \end{aligned} \tag{66}$$

where 'move', 'attr', 'entered net' and 'unphys jump' are defined in Equations (34), (35), (37) and (41) respectively. If the coordinates gained inside the net is accepted, we switch the direction of the net repulsion, as the mosquito moves towards the host that is under the bed net as modeled by Equation 49. Also, the candidate step gained outside the hut is accepted with probability, p_{hut}, of which we assume that the mosquito has entered into the window traps. In the case of rejection, we simulate the mosquito hitting the wall and update the mosquito's new position $\mathbf{x}^n$ to be the closest point on the wall, than that of position $\mathbf{x}^{n-1}$ such that $\parallel \mathbf{x} - \mathbf{x}^h \parallel = h_s$, where h_s is the hut size as shown in Table 5.

Thus we mark the mosquito hitting the wall as

$$\text{condition}(\text{ind\_hit } \& \ \sim \text{acc\_hut}, 11) = 1, \tag{67}$$

where ind_hit and acc_hut are modeled as

$$\begin{aligned} \text{ind\_hit} &= \text{move } \& \text{ attr } \& \text{ exited hut}, \\ \text{acc\_hut} &= u < p_{hut}, \end{aligned} \tag{68}$$

where ‘move’, ‘attr’ and ‘exited hut’ are defined in Equations (34), (35) and (39) respectively. However, a mosquito which has been marked as hitted can exit the hut if acc_hut is achieved.

The table below presents the formulas of the models formed in this subsection and the names of the functions/scripts for which they are executed in MATLAB.

Table 4. Trace back to MATLAB.

Model executions	**MATLAB script/function names**
Equations (65) and (66)	net_barrrier6.m
Equations (67) and (68)	hut_barrier6.m
Parameter usage	hut_exp_init.m .

3.5 Model Algorithm

The algorithm for implementing the models are summarized in Algorithm 3.

Algorithm 3 Model Algorithm

1. Select candidate point $\mathbf{x}^n$ by adding a stochastic increment to the previous point, $\mathbf{x}^{n-1}$, as given by Equation 24;
2. Account for mosquito mortality by computing the total probability of death using Equation 59; generate random number $u \sim U[0,1]$ and remove the agent if $u < \alpha^{\Delta t}$;
3. Measure the concentration of attractive odour $C(\mathbf{x}^n)$ at new position $\mathbf{x}^n$ as described in Equation 28;
4. Compute the scaling factor $\sigma_{acc}(\mathbf{x}^{n-1})$ as given in Equation 45;
5. Compute the probability of accepting a new position $\mathbf{x}^n$ influenced by attractive odour; $\alpha_a(\mathbf{x}^n|\mathbf{x}^{n-1})$, as specified in Equation 30;
6. Compute the repellent probability; $\alpha_r(\mathbf{x}|d_{50}, r, s)$ by Equation 48;
7. Account for current candidate position by checking if the criteria described in Equation 35 is met of which such mosquito's new position $\mathbf{x}^n$ is as primarily accepted; otherwise, we mark it as rejected such that it remains at old position by assigning $\mathbf{x}^n = \mathbf{x}^{n-1}$;
8. Account for net barrier. If a candidate step $\mathbf{x}^n$ is inside the net and the old position $\mathbf{x}^{n-1}$ is outside and $\mathbf{x}^n$ was primarily accepted, generate random number u. If $u < 1 - p_{net}$, accept the new position $\mathbf{x}^n$. Otherwise, choose the closest point on the net $\mathbf{x}^{net}$ than $\mathbf{x}^{n-1}$ and assign $\mathbf{x}^n = \mathbf{x}^{net}$;
9. Account for walls. If a candidate step $\mathbf{x}^n$ is outside the hut and the old position $\mathbf{x}^{n-1}$ is inside and $\mathbf{x}^n$ was primarily accepted, generate random number u. If $u < p_{hut}$, accept the new position $\mathbf{x}^n$. Otherwise, choose the closest point on the wall $\mathbf{x}^{wall}$ than $\mathbf{x}^{n-1}$ and assign $\mathbf{x}^n = \mathbf{x}^{wall}$;
10. Update the relevant property lists of mosquitoes that are not dead with the respective formulas as given in Table 4 ;
11. Score fed mosquitoes by considering the formula given in Equation 61; where epsilon is the minimal distance between the mosquito and host, which is treated as an exposure.
12. Start again from step 1 and set $n \rightarrow n + 1$.

We repeat the algorithm steps so that the time period of one night, 10 hours is covered,

and also taking the additional delayed mortality into account. Thus, each calculation simulates a 34-hour hut experiment. In the simulation, one iteration step corresponds to 2 seconds and there are $18,000$ iterations. The algorithm was implemented in MATLAB and the graphical user interface (GUI) was helpful in implementing the visualization of the experiment.

3.6 Model Parameters

Given that some of the parameters employed in the above described models are related to given physical factors, we fix such kind of parameters for all simulations. Thus the table below gives the values and description of these parameters.

Table 5. Fixed parameters

Parameter symbols	**Parameter description**	**Parameter values**	**Source**	
μ_s	Average flight speed	0.4 m/s	[36]	
σ	Standard deviation of the fight speed	0.1 m	[42]	[36]
$3\sigma_a$	Maximal distance at which mosquito is able to sense a host	80 m	[45]	
h_s	Size of the experimental hut	3 m	[46]	
d_p	Net width	1.5 m	[46]	
ϵ	Minimal distance between mosquito and host considered as an exposure	0.65 m		
t_{max}	Maximal host-seeking time in the absence of chemicals	5 h	[47]	
s	Slope of the repellent which characterizes the spatial spread of repellent	0.01	[48]	
s_μ	Slope of the chemical that characterizes the temporal range of effective consumed dosage	0.001		
σ_μ	Effective range of the poison	≤ 10 cm	[48]	

Furthermore, we can fix the parameters, $\sigma'_{acc}, \sigma''_{acc}$ of the scaling factor, σ_{acc}, by the following argument. We recall that outside the concentration plume (i.e at a distance of 80m), the mosquito movement is purely random. This distance is never reached in the hut

experiment; considering the size of the experimental hut. The upper limit for σ_{acc} was taken large enough to produce Brownian movement outside the CO_2 plume. Thus, using Equation 45 and accounting for $\sigma' << 1$, we have that with

$$\sigma_{acc}(\mathbf{x}| \parallel \mathbf{x} - \mathbf{x}^h \parallel = 80) = 0.001, \tag{69}$$

the second attraction parameter can be given as; $\sigma''_{acc} = 0.001/80$, where the value, 0.001 is given by the difference in the concentration values, $C(\mathbf{x}^n, \mathbf{x}^h) - C(\mathbf{x}^{n-1}, \mathbf{x}^h)$ at the distance of 80m from the host. The value of σ'_{acc} should be identified by parameter estimation given the upper limit of σ_{acc}. From the range of possible values, considering the upper bound for σ'_{acc} given in [22], the choice of $\sigma'_{acc} = 0.0001$ was made.

Hence, with these stated values, the acceptance probability for steps away from the host tends from one to nearly zero when the distance of the mosquito from the host, $\parallel \mathbf{x} - \mathbf{x}^h \parallel$ decreases from 80m to zero which implies a transition from a pure random walk outside the concentration plume to increasingly and greedy movement at a short distance from the host as illustrated in Figure 5. Other parameters were identified from real data using adaptive MCMC method.

The parameter chains generated with MCMC are used in the algorithm differently for control and treatment case. In the control case, parameters ruling mosquito attraction to the host were fixed and variable parameters; probability of penetration through the net, $1 - p_{net}$ and probability of exiting from the hut, p_{hut}, were sampled from previously generated MCMC chains at each iteration of the algorithm. In the treatment case, parameters defining the net repellent, insecticide-induced death rate and insecticide-induced exit rate were sampled from the chains on each iteration of the algorithm while the maximum host-seeking time in the presence of repellent is fixed to the same value as in the control case. This is done in addition to the control case parameters.

3.7 Assumptions of the Models

Having discussed the modelling approach, we make the following assumptions for the simulation;

(i) The mosquitoes feed only on human and do not have any alternative source of blood meal.

(ii) Since simulation time is relatively short and considering that it usually takes time for mosquitoes to digest their blood meal after feeding, we assume that sufficient amount of blood meal is obtained after one feeding. To that end, each mosquito takes only one bite after which the mosquito agent switches to a pure random walk (non directional movement).

(iii) There is no difference in the mosquito preference from one person to another.

4 ANALYSIS OF MODEL RESULTS

In this section, we provide a means of getting statistically reliable results in respect of the stochastic nature of the model outputs. Next, a sensitivity analysis is conducted for the control case model and then, both the control case and treatment model are finally calibrated with several real data sets from experimental hut trials.

4.1 Uncertainty Quantification

The adequacy of any model is based on the correctness and reliability of its output. However, since all models are imperfect simplifications of reality, and also due to the fact that the obtained data with which the models are calibrated are noisy, the output values are subject to imprecision. Thus, we outline some sources of imprecision in model outputs as we pinpoint the source inherent in our model and thereby give ways of quantifying, and communicating the uncertainties in the outputs.

4.1.1 Sources of Uncertainty

Imprecision and uncertainty associated with model outputs could be as a result of numerous reasons. The various sources of uncertainty are as described below;

- Error in obtained measurement. Also known as observation error, this comes from the variability of experimental measurements.

- Imperfect representation of processes in a model (model structure) as compared to the real system alongside the approximations made by numerical methods employed in the simulation.

- Paucity of knowledge of the essential features (parameters) that the model require to portray reality.

- Imprecision in specifying the values of parameters associated with the model structure. If the model calibration process is repeated while employing different data sets, different parameter values would result. These values would produce different simulated model behaviors, yielding different model outputs in return.

- Variability emanating from observed input and output values over a region and within a time that differ from the spatial and temporal scale of the model.

- Errors in the model solution algorithm.

- Uncertainty as a result of the natural randomness in a process.

According to [49], the above described sources of uncertainty are classified under two broad categories as discussed below.

- **Epistemic uncertainty** also known as systematic uncertainty, and is as a result of some things that one could know in principle, but doesn't in practice. These sources of uncertainty are reducible and can be evaluated by gaining better knowledge of the mechanism under study.

- **Aleatory variability** also known as statistical uncertainty, is the inherent randomness in a process which usually stems from the unknowns that differ each time the same experiment is ran. The quantification for the aleatoric uncertainties can be relatively straightforward to perform, for instance by using Monte Carlo techniques and thereby estimate the mean and standard deviation of the model outputs.

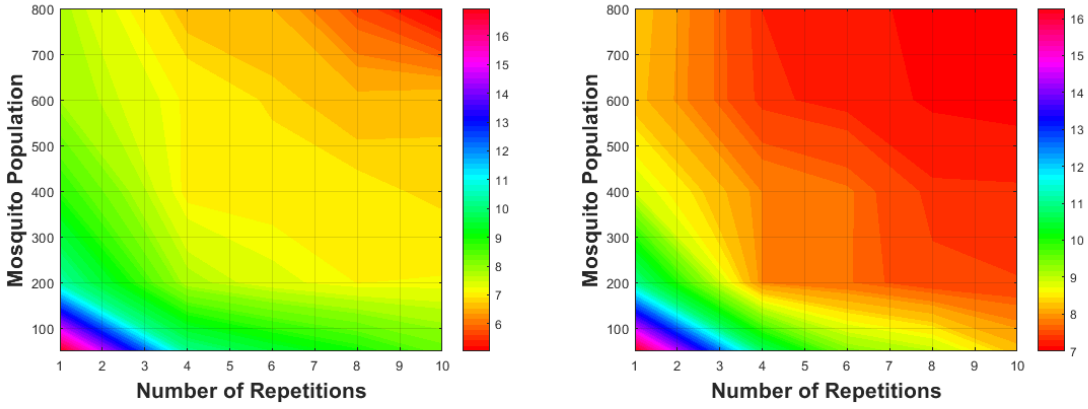
This study addresses the uncertainty quantification of the model outputs of which source of uncertainty, involves the notion of randomness.

Given that the simulations are stochastic due to the random numbers generated during the simulations (among which include random numbers generated for: selecting dead mosquitoes, candidate steps and accept/reject) and also as a result of the fact that some of the model parameters employed in the experiment, are not well known but are identified with the help of sampled MCMC chains, we have that the model outputs are stochastic. Thus, in order to calibrate the model against real measurements, by estimating the impact of various model parameters, we need the averages of the model outputs. This is done by repeated simulations using a given number of mosquitoes in each. To get a statistically reliable result, we study the variability of the resulting outputs so as to get a standard deviation that is low enough to be reliable; using the standard deviation in the data employed for calibration, as the threshold for reaching to a plausible conclusion.

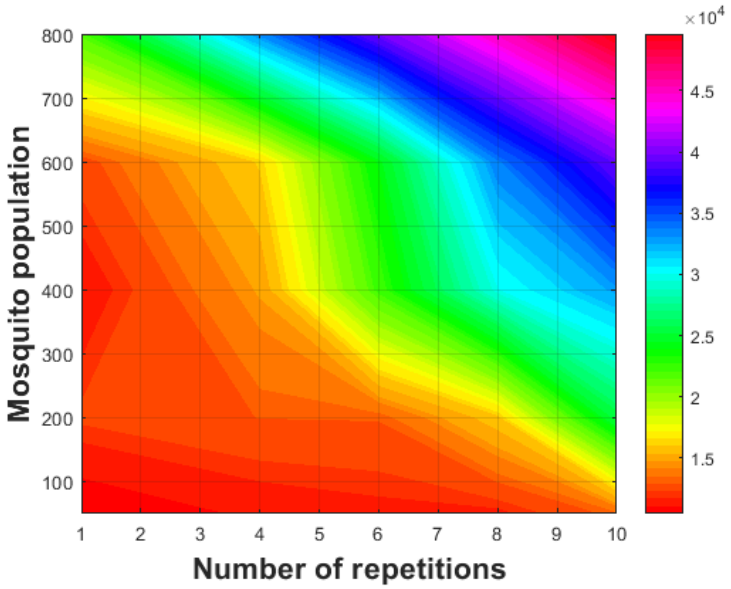
The standard deviation of the simulations can be reduced in two ways. First, by taking larger number of mosquitoes and next, by taking the average of repeated simulations. Given that the repeated simulations for swarms of mosquitoes is really CPU intensive to

carry out, minimizing the computational times is of paramount importance. Thus, we seek for a compromise between the number of mosquitoes and the number of simulation repetitions that minimizes the computational times and still keeps the standard deviation for simulations small enough within the aforementioned threshold.

In doing this, we arbitrarily and constructively considered $50, 200, 400$,600 and 800 mosquito populations with $1, 4, 6, 8, 10$ number of repeated simulations for averaging applied for each population size. A constant number of 20000 iterations are taken for each case after-which we check the standard deviations of the model results based on each computation of the cost function in MCMC runs. This was done for only the control case in this study as two factors were considered; percentage exit rate and percentage fed rate of mosquitoes. Figure 8 gives the visual results depicting how the standard deviation varies with respect to the number of repeated simulations and mosquito population size alongside the computational time for each combination.



(a) (b)



(c)

Figure 8. Results of uncertainty quantification. (a) Standard deviations of percentage exit rate. (b) Standard deviations of percentage fed rate. (c) Computational time (in seconds).

Overall, it can be seen that that there is a clear difference in the variation of the model outputs with respect to the population size and simulation repetitions for averaging as standard deviation decreases with increase in the population size and the number of simulation repetitions. The computational time as expected, increases with increase in both the mosquito population and the number of simulation repetitions.

Considering the standard deviations of the data presented in [22], calculated at 95% level of confidence and are being used as threshold for choosing a small enough standard deviation, and taking the computational time to cognizance, we reach a compromise of

600 mosquitoes at 4 simulation repetitions. Thus, the computational time for such a run is 16463.92082 seconds (approximately 5 hours) using DESKTOP-JS2F88A, intel(R) core(TM) i7-6700HQ CPU @ 2.26GHz. The GPU card that was employed for computations is NVIDIA Get Force GTX 980M.

4.2 Sensitivity Analysis

A sensitivity analysis which seeks to determine the changes in model output values as a result of modest changes in model input values [50] was conducted for the control case model. Changes in the values of model parameters can impact the values of model outputs in diverse ways. It should be noted that, a relatively few input variables substantially influence the values assumed by a particular output variable. Hence, considering that the range of uncertainty of only two of the model outputs (percentage exit and percentage fed mosquitoes) is of interest, then undoubtedly only those parameters that significantly influence the values of the aforementioned responses are included in the sensitivity analysis. These parameters are considered to be p_{net} and p_{hut} which denote the probability of being blocked by a net barrier and the probability of exiting the hut into the window traps respectively. Based on the model, we considered the range of values of p_{hut} and p_{net} to be from 0.999 to 0.9999 and 0.0001 to 0.01 respectively which were taken as inputs. Thus, the results obtained are showcased in Figure 9 for percentage exit rate and percentage fed rate, dependent on the two parameters considered.

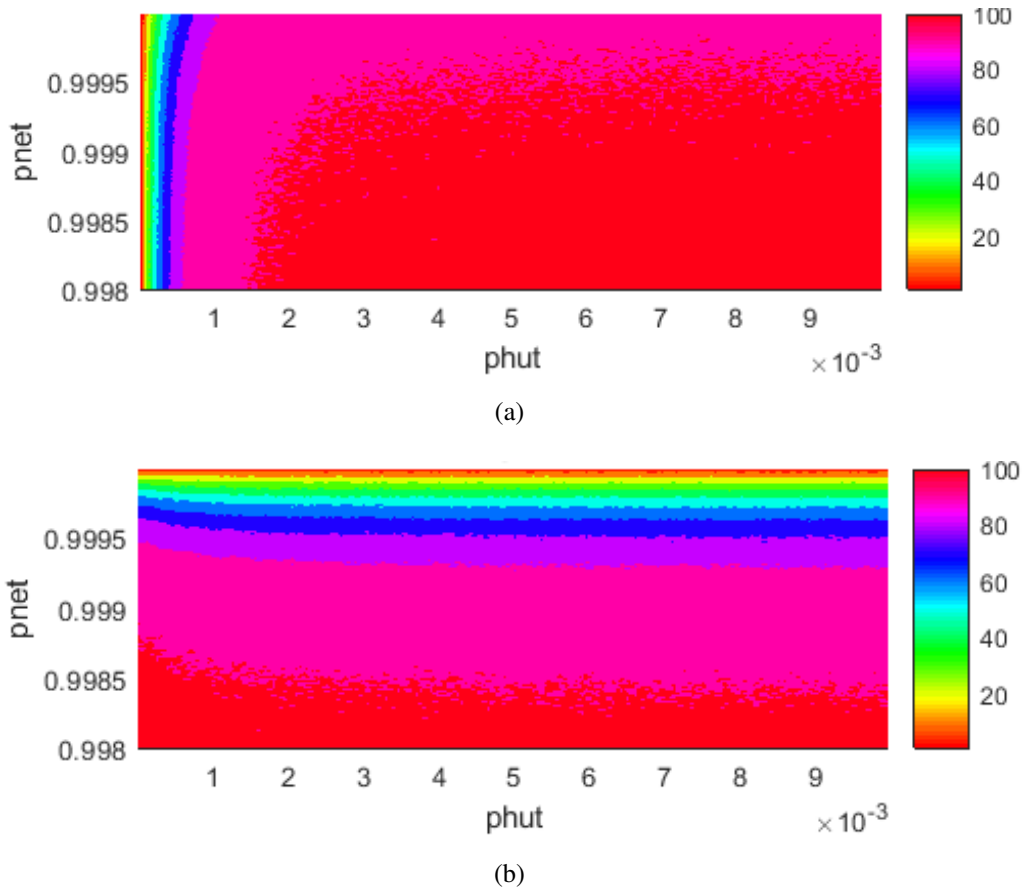


Figure 9. Plots illustrating the effects of input parameter values on the model outputs. (b) Percentage exit rate. (b) Percentage fed rate.

It can be seen from the above plots that p_{net} affects fed rate and has very little impact on exit rate. Same thing is applicable to p_{hut} as its effect is greatly noticed in the exit rate and has little impact on fed rate. The following relations hold as regards the impact of the two parameters on the model outputs.

- Increase in p_{net} brings about a decrease in the percentage rate of fed mosquitoes and slightly reduces the exit rate. Intuitively, we have that increasing the probability for which a mosquito is being blocked by a net barrier (either by reducing: the size of the holes, the number of holes in a net or both ways), causes the mosquito to stay longer in the hut in search of a hole to penetrate and this reduces the rate at which they exit.

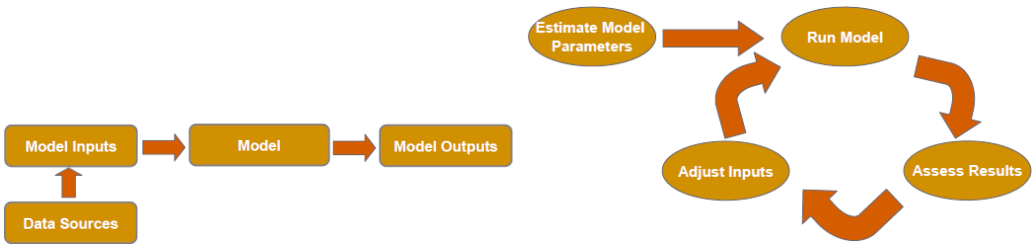
- Increasing p_{hut} makes the percentage rate of mosquito that exit from the hut to increase and brings about a slight decrease in the percentage of the fed mosquitoes. An instinctive interpretation to this is that increasing mosquitoes' chance to leave

the hut can make the rate at which they loose interest in search of a host to decrease especially for An. arabiensis which is known to have alternative sources of blood meal other than the human host.

Furthermore, some MCMC sampling was made to check how the model fits the data from [22] with certain parameter values. The graphical presentation of the parameter values that make the model fit the data within reported error bounds, in respect of data uncertainty is as shown in the Figure 11.

4.3 Model Calibration and Parameter Identification

Calibration involves the estimation of the values of various parameters in a given model, in order to fit a given set of data [51]. Thus, the calibration process is a trial and error effort that seeks the parameter values which have the greatest likelihood of being accurate within acceptable error tolerance. Upon obtaining satisfactory estimates of the model parameters, the models are checked to ensure that they adequately perform the functions for which they are intended; which in our case, is to accurately estimate the percentage exit, fed and mortality rates of the mosquito species under study. The overview of the model calibration process is given in Figure 10.



(a) (b)

Figure 10. Model calibration process (source:[51])

From the illustration obtainable in the above figure, we have that an initial guess of the parameter values are obtained and the model is thereby run, to obtain simulated data values that are compared with the corresponding observations. The process is repeated until a satisfactory correspondence is obtained and this is what is obtainable from least squares fitting which seeks for one best parameter estimate that reduces the sum of squared difference between the real data and the model outputs.

Considering that there are uncertainties in the observed data sets, which are given in terms of confidence regions, we therefore regard the parameters as random variables and seek for the distribution of all model parameters that fit the data with regards to the obtainable level of noise in the measurements. Furthermore, another concept of interest is to know the extent at which the various model parameters actually can be identified by the available real data and this is where parameter identifiability comes into play.

A suitable head-way in that respect, is given by the MCMC approach which as described earlier in Section 2, involves drawing candidate parameter values from a predefined proposal distribution and these candidate points are either accepted or rejected based on some level of likelihood in the sense that the closeness of the model output to the data is taken into consideration. As there are many MCMC methods that can be employed as discussed in Section 2.2, Adaptive MCMC is particularly employed in this study since at the initial instant, we may not be able to clearly identify a well working proposal distribution. The MCMC technique allows the creation of possible combination of parameter values which fit the data within a given tolerance for the measurement noise. Thus, we calibrated the model for each data set by employing the aforementioned MCMC technique with a chain length of at least, 20,000 using a swarm of 600 mosquitoes and 4 repetitions which are averaged for a given sample. Considering that the task involves extensive computational sampling methods, the computations were carried out by combining MATLAB and GPU programming which suits the simulation of each agent independently. Thus, a CUDA code is made which is called from MATLAB by creating and running a kernel in C CUDA and additionally a mex function, wrapped around the CUDA code is created. The interface for compiling the dynamic library which resulted from compilation of the CUDA project into mex was implemented in C language. The model of the GPU card that was employed for these computations is NVIDIA Get Force GTX 980M. However, the CPU optimization carried out in Section 4.1 is helpful to further reduce the computational time.

The cost function which gives the sum of squared difference between the observations and the model outputs while accounting for the measurement error variance, is used to evaluate the fit with the data. Thus we have

$$\mathbb{S}sum = \sum_{i=1}^{N_r} \frac{(Y_i - \hat{Y}_i)^2}{\sigma_i^2}, \tag{70}$$

where N_r is the number of responses over which the sum of squared is computed, Y_i and $\hat{Y}_i$ denote observed data and the simulated model outputs respectively. Furthermore, in conformance to the confidence bounds given in the various data sets in which our model

was calibrated, we roughly allow a certain percentage rate of variation in the model output by setting the measurement error variance as calculated from the confidence bounds reports in the respective literatures.

4.3.1 Control Case

In the control case, the experimental hut trial with an untreated net is considered, under the assumption that both species of mosquito exhibit similar host-seeking behaviors in the absence of insecticide-impregnated bed net. To that end, the same parameters are applicable to both species in this case.

Recalling from Table 5 that the maximum host-seeking time in the absence of any repulsive or poisoning effect is fixed at 5 hours, we then calibrate two more principal parameters for the negative controls: probability of being blocked by just a physical barrier posed by the untreated net (p_{net}) and the probability of exiting the hut (p_{hut}). Hence, we fit these parameters in consideration of two measured factors (percentage exit rate and percentage fed rate) from [22], [52], [53] and [54] such that $N_r = 2$ in Equation 70. The datasets with which the hut experiment model for the control case is calibrated, were measured virtually under the same experimental conditions and procedures and also conforms with the structure of the model. Table 6 gives the initial parameter guesses for the MCMC sampler with respect to the two variable parameters.

Table 6. Model parameters for the control case, Initial guesses for the sampler and the standard deviation allowed with respect to each dataset.

Parameter symbols	Parameter description	Initial values				
		1	2	3	4	5
p_{net}	probability of being blocked by a net barrier	0.99978	0.999699	0.999899	0.9999	0.999499
p_{hut}	probability of exiting the hut	$6.1 \cdot 10^{-4}$	$4.9 \cdot 10^{-4}$	$1.0 \cdot 10^{-3}$	$7.1 \cdot 10^{-4}$	$3.9 \cdot 10^{-4}$

The results of the parameter estimation are shown in Figures 11 to 15.

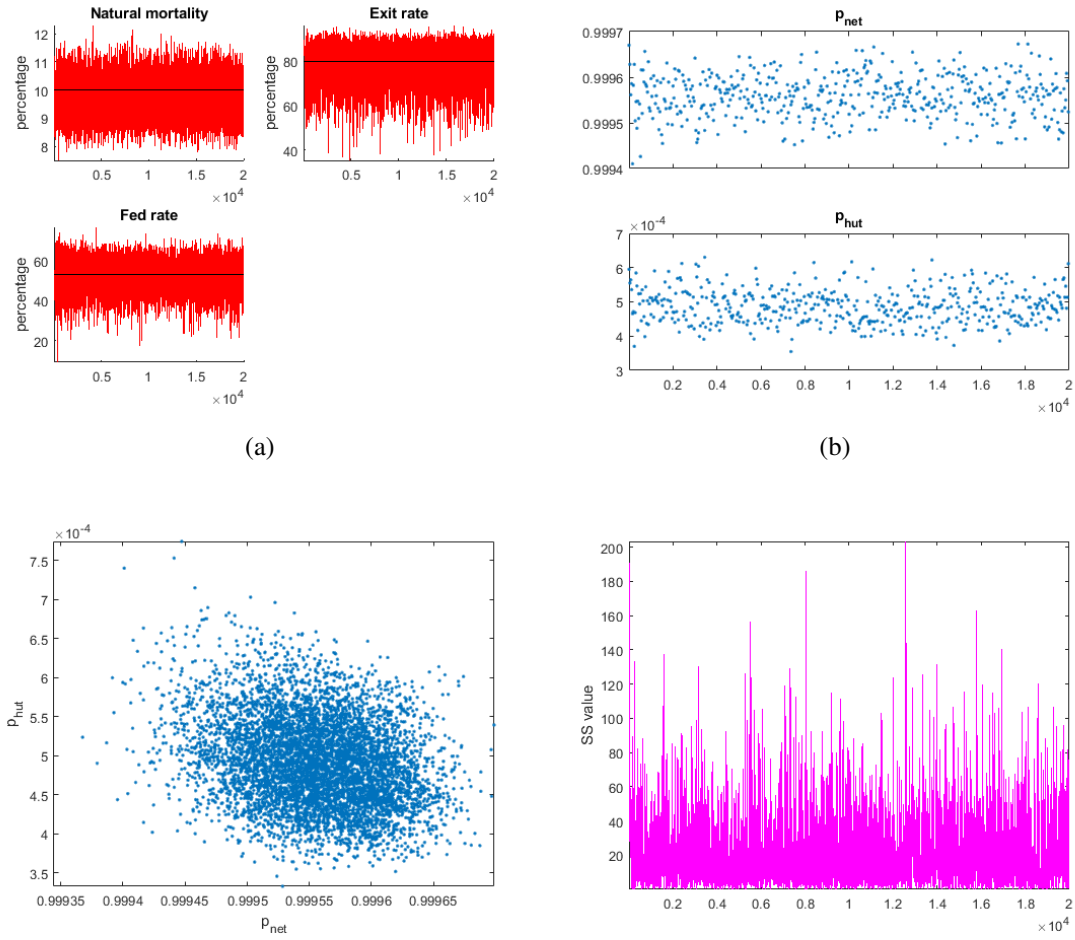


Figure 11. Results of model calibration with data from [22]. (a) Model outputs vs measured data from [22]. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

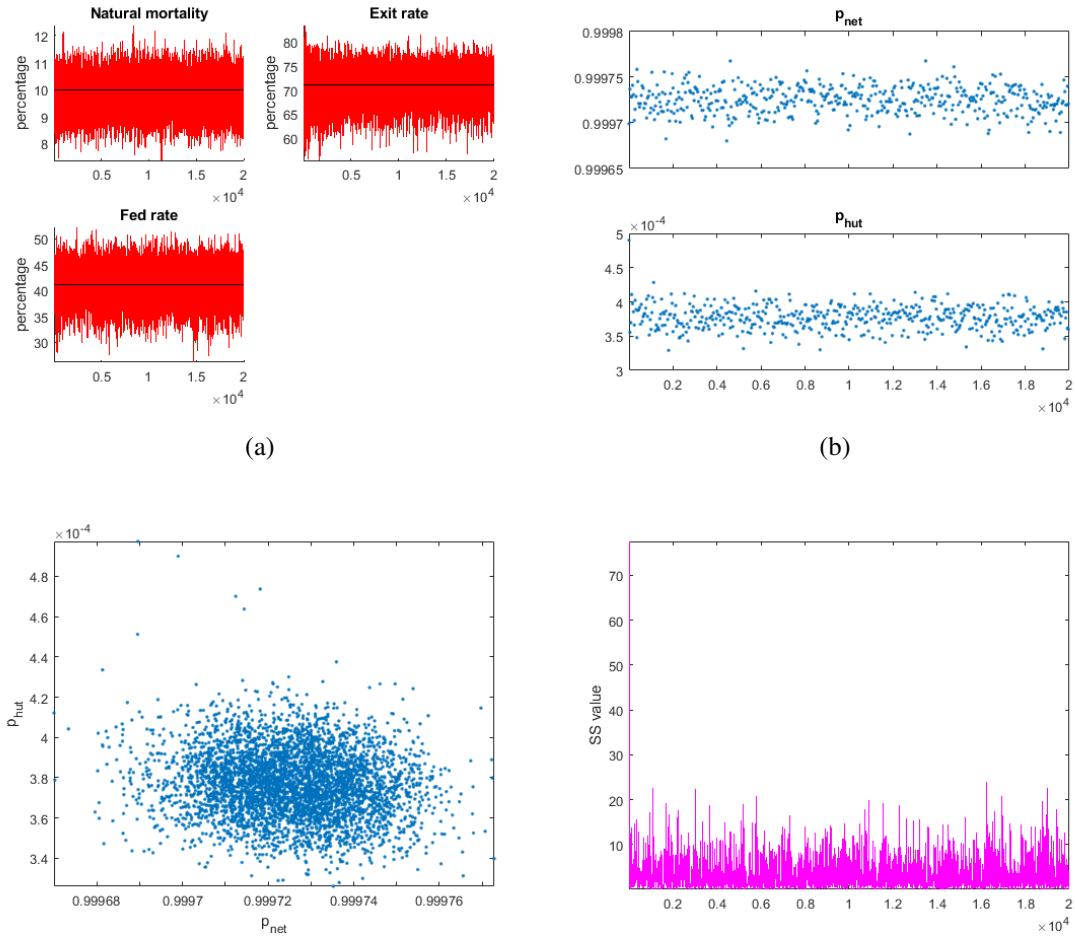


Figure 12. Results of model calibration with data from [52]. (a) Model outputs vs measured data from [52]. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

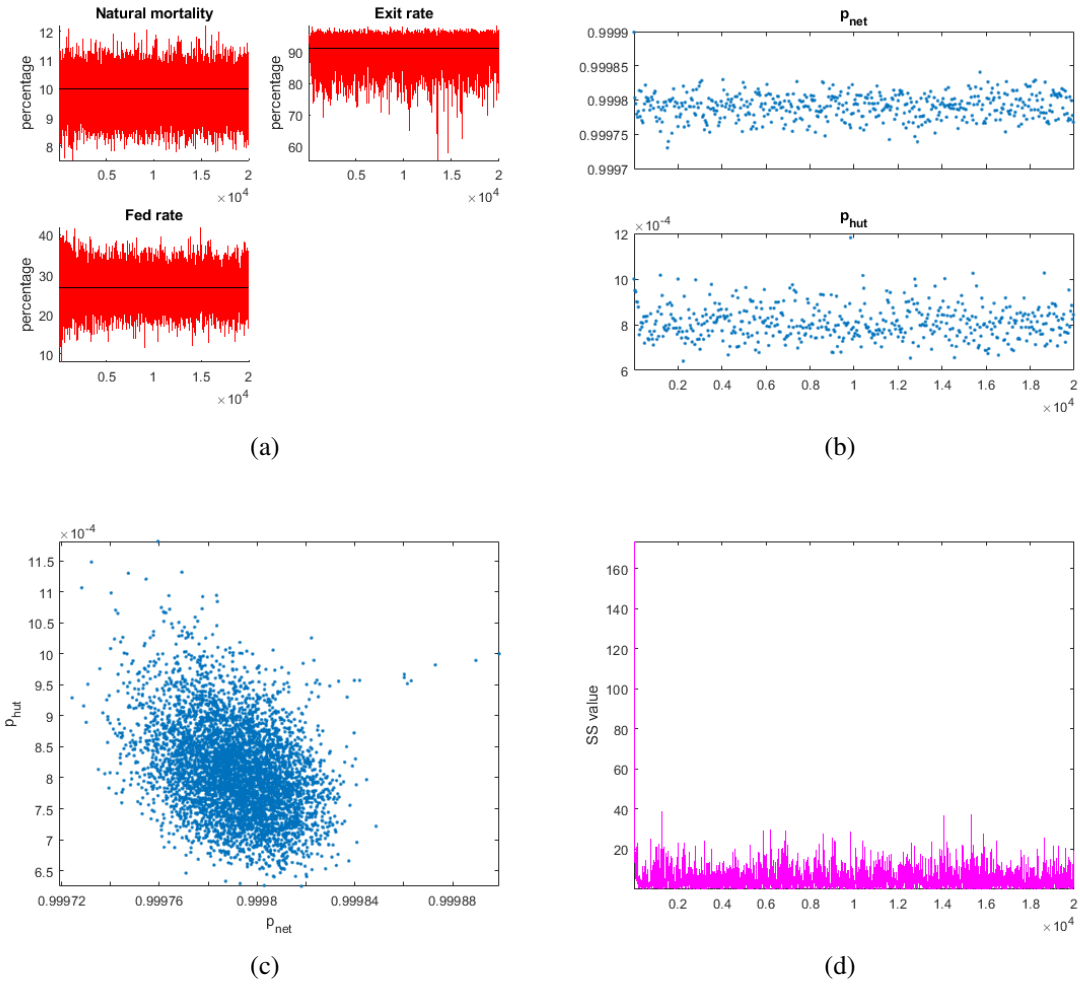


Figure 13. Results of model calibration with data from [53]. (a) Model outputs vs measured data from [53]. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

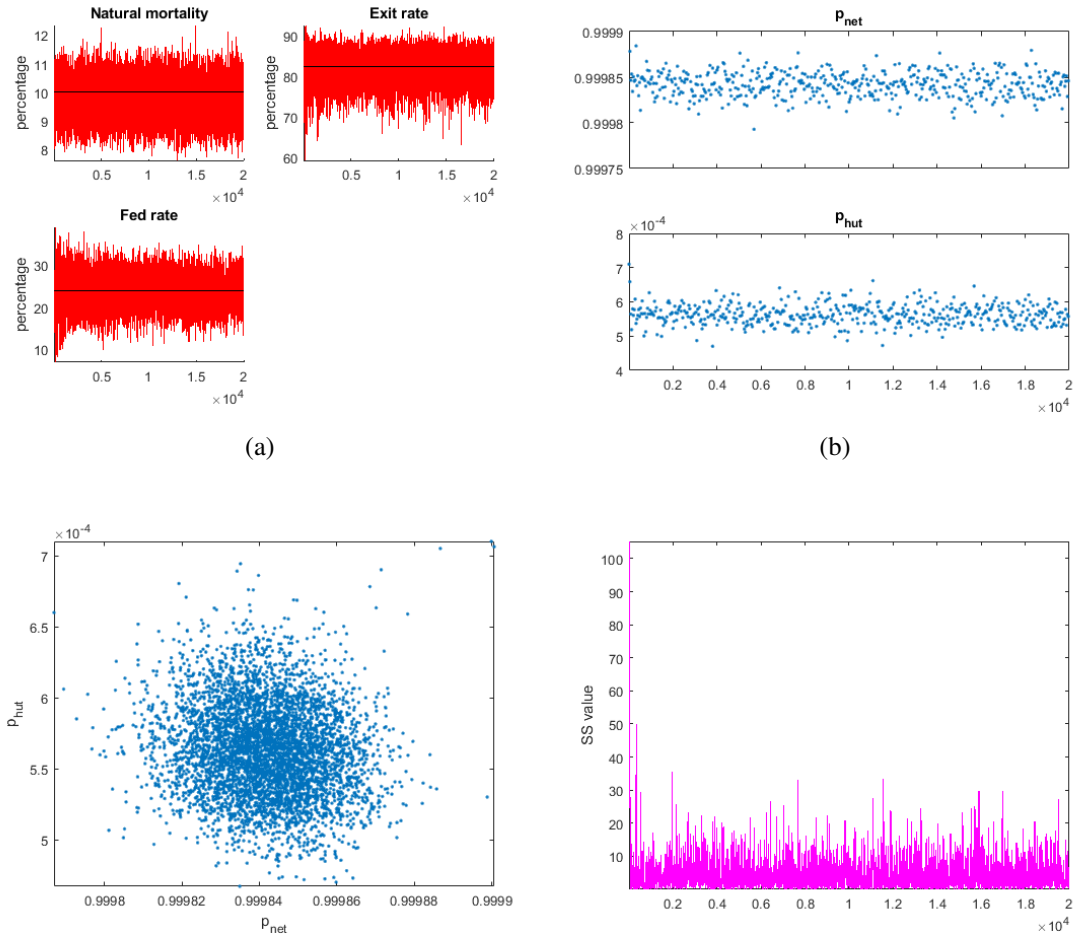


Figure 14. Results of model calibration with the first data from [54]. (a) Model outputs vs measured data from [54]. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

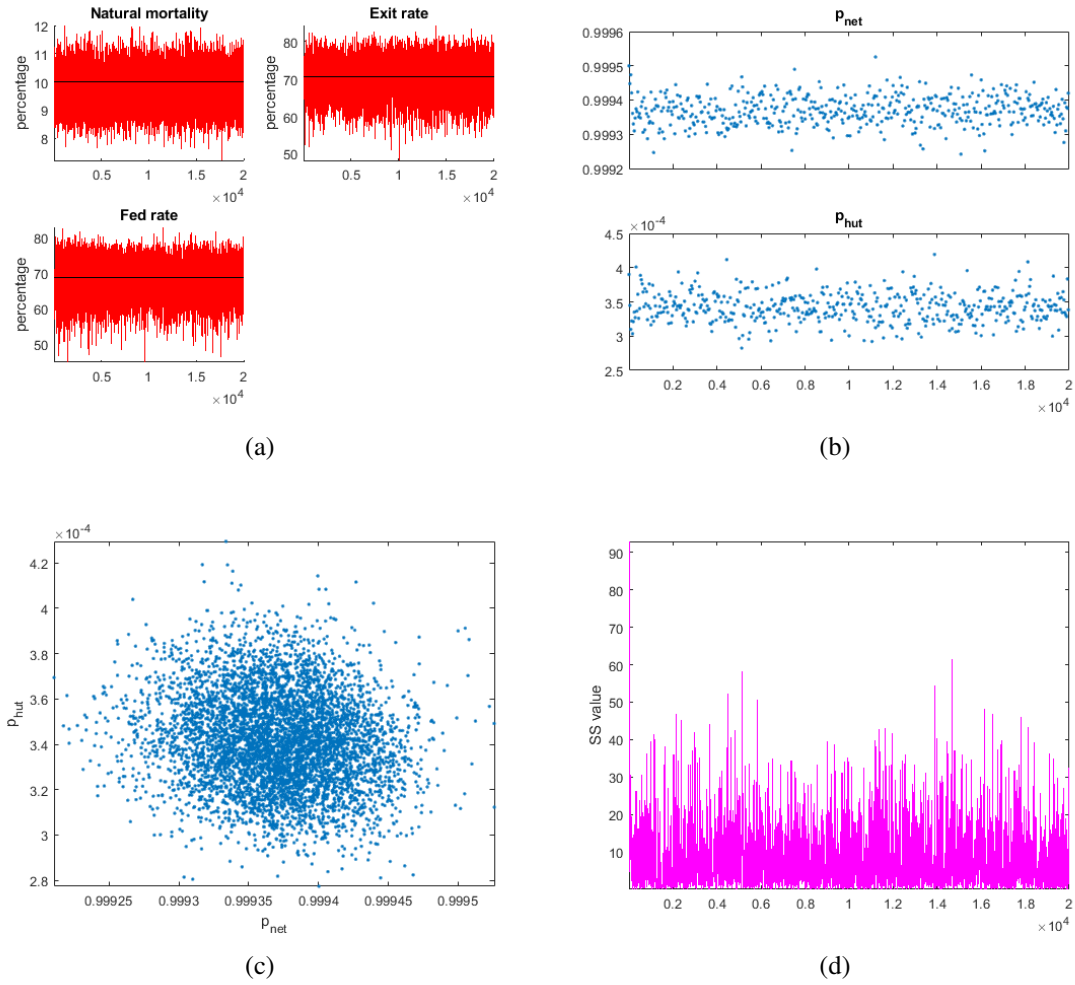


Figure 15. Results of model calibration with the second data from [54]. (a) Model outputs vs measured data from [54]. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

Overall, it can be seen from the above results that the model fits all the measurements real good as shown in the plots of the simulated model outputs (red trace lines) against the real measurement represented by the constant black lines. This is evident in the sum of squared chain plots which show that the the difference between the model outputs and the measured data tends to approximately zero at most points. However, it should be noted that the outcome of the sum of squares partly depends on the given estimate for the experimental error. On the other hand, the variability of the simulated model outputs has a close match with the range of data uncertainty reported in the various literatures.

Furthermore, the distribution of the model parameters that produced the model fit to the various data sets are given in Figures 11b, 12b, 13b, 14b and 15b and it can be seen that

all parameters are well identified by the available real data. On the side of the joint 2D plots, it can be seen in all cases that there is no significant correlation between p_{net} and p_{hut} which conforms to the underlying expectation, with regards to their functionalities as explained in the sensitivity analysis in Section 4.2. Thus, the credibility of the hut experiment model for the control case has been ascertained by demonstrating its ability to replicate actual control case data sets.

4.3.2 Treatment Case

The treatment case model, unlike the control case model where it is assumed that both *An. gambiae* and An. arabiensis exhibit similar host-seeking behavior, seeks to calibrate the behavioral differences in host-seeking of these two species upon confrontation with ITN/LLIN.

According to the data from the hut experiments conducted using nets treated with different chemicals in [22], the mortality rate of *An. gambiae* is substantially and consistently higher than that of *An. arabiensis* and the exit rate in most cases is lower in *An. arabiensis* than *An. gambiae* whereas, insecticide induced exiting was relatively higher for *An. arabiensis*. On the other hand, the data also revealed that *An. arabiensis* features higher feeding rate than *An. gambiae* (except for IconMaxx LN).

While several parameterization have been adopted to capture the scenario inherent in the treatment case data sets in [22] as discussed in [23], it turned out that the fits obtained are good enough only for the IconMaxx chemical data for which unlike in the datasets given in [22], the death rate of *An. gambiae* was not up to twice higher than that of *An. arabiensis*. Thus, we have that the fits obtained therein are not ideal since the model is not a good fit to the other sets of data in [22]. Explicitly, the fact that *An. arabiensis* has higher (or equal) feeding rate than *An. gambiae* and considerably lower death rate, contradicts with the mechanism of the model inherent in [23] since both the probability of death and that of successful feeding is proportional to the number of net contacts and as such, it is not pragmatic to simultaneously have high feeding rate and low mortality rate for a mosquito specie.

A number of probable reasons can be offered to account for the situation inherent in the data. One of such explanations is that the rate of poisoning is different for these two species such that, it takes time for the poison to get from the salivary glands to the neural system of mosquito and this time delay is suspected to be different for the two mosquito

species. However, considering that the bioassys in [22] state that a big dosage is equally lethal for both *An. gambiae* and *An. arabiensis* but mosquitoes do not acquire the lethal dosage upon a single contact with the net but rather a sub-lethal dosage, we postulate that the concentration of chemical decays with time and at some rates which is different for this two species [55]. In order to account for the detoxification rates in the mosquito species, we introduce two parameters which accounts for chemical detoxification in *An. gambiae* and *An. arabiensis* respectively such that with reference to Equation 51, we model the dynamics of chemical concentration with the formula,

$$\frac{d\mathrm{D}(\mathbf{x}, \mathbf{x}^n)}{dt} = -\alpha \cdot \mathrm{D}(\mathbf{x}, \mathbf{x}^n) + \mathrm{source}(t), \tag{71}$$

where α is the detoxification rate which depends on the chemical and mosquito specie and source, is the dosage of chemical which the mosquito could contact at time t, otherwise, it is zero. Moreover, it can be noted that there is a parameter μ_p which rules the impact of chemical poison and common for both species and it is known as the poisoning rate. For the results presented in the present work, we employed the linear poisoning function in computing the chemically induced death rate, parameterized as

$$\mu_c = \mu_p \cdot \mathrm{D}(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n, \mathbf{x}_N), \tag{72}$$

where $\mathrm{D}(\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n, \mathbf{x}_N)$ is the total accumulated dosage which is dependent on the sequence of positions and the poisoning rate, μ_p is fixed at 1.

Furthermore, in order to properly fit the exit rates, two parameters which denote the rate of exito-repellency respective for both species [56], are employed while keeping the host-seeking time t_{max} of both species fixed to the same value as in the control case. The exito-repellency is modeled as

$$\sigma_{acc}(\mathbf{x}, C_{rep}) = \sigma_{acc}(\mathbf{x}) + \mu_e \cdot C_{rep}, \tag{73}$$

where μ_e depends on the mosquito specie and also on the insecticide, C_{rep} is the total number of steps taken by the mosquito in the repellent plume and $\sigma_{acc}(\mathbf{x})$ is the linear distance dependent scaling factor introduced to account for the increasing greediness of mosquito in the odour plume at short distance to the host as in Equation 45. Thus, Equation 73 presents a scaling factor which is conditioned on the distance and the repellent effect. To that end, we have that a higher rate of increase in exito-repellency, μ_e will bring about an increased value of the scaling factor, thereby giving the mosquito a greater chance for its steps away from the host to be accepted. However, the parameters for exiting the hut and penetrating the net are kept the same as in the control case. Accounting for

the impact of repellents, we employ the function given in Equation (48) where d_{50} is fixed at 0.75 and the repulsion intensity, r is estimated for different insecticides by MCMC.

Thus, we fit these parameters in consideration of six measured responses ($N_r = 6$ in Equation 70): percentage exit rate, percentage fed rate and percentage mortality rate (corrected for control by 10%); all of which are separate for both *An. gambiae* and *An. arabiensis*. The data on experimental hut trials with bednets treated with IconMaxx LN, Deltamethrin, Alphacypermethrin and Olyset kit [22], are employed in calibrating the model. Table 7 presents the initial parameter guesses for the MCMC sampler with respect to the supposed five variable parameters.

Table 7. Model parameters for the treatment case, Initial guesses for the sampler and the standard deviation allowed with respect to each dataset.

Parameter symbols	Parameter description	Initial values			
		IconMaxx	Deltamethrin	Alphacypermethrin	Olyset
r	Repulsive strength of the ITN/LLIN	0.800	0.000	0.500	0.000
α_G	Detoxification rate for *An. gambiae*	$8.25 \cdot 10^{-4}$	$3.78 \cdot 10^{-4}$	$7.25 \cdot 10^{-4}$	$8.06 \cdot 10^{-4}$
μ_e^G	The rate of increase of exito-repellency in *An. gambiae*	0.8	$3.00 \cdot 10^{-5}$	1.00	1.00
α_A	Detoxification rate for *An. arabiensis*	$1.10 \cdot 10^{-3}$	$1.20 \cdot 10^{-3}$	$2.40 \cdot 10^{-3}$	$2.60 \cdot 10^{-3}$
μ_e^A	The rate of increase of exito-repellency in *An. arabiensis*	$1.00 \cdot 10^{-2}$	$7.39 \cdot 10^{-6}$	$6.00 \cdot 10^{-7}$	$9.00 \cdot 10^{-6}$

Owing to the fact that no formal sensitivity analysis has yet been conducted for the current treatment case model parameters, we however give a description of what is obtainable from the parameters below.

- Reducing the repulsive strength r of a net brings about a reduction in the exit rate and increases the feeding rate for both mosquito species and vice versa.

- Increasing the poisoning rate, μ_p causes an increase in the death rates of both mosquito species.

- The higher the poison detoxification rate in any specie, the lower their death rate alongside their exit rate and the higher the fed rate of that specie and vice versa.

- The higher the rate of increase of exito-repellency, in any specie, the higher the exit rate and the lower the death rate and fed rate of that specie and vice versa.

However, it should be noted that each of these parameters affect the measured responses at different rates but a detailed explanation can be rendered by conducting a typical sen-

sitivity analysis. The model calibration results with data on hut trials, employing bednets treated with IconMaxx LN, Deltamethrin, Alphacypermethrin and Olyset kit are presented in Figures 16 to 19 respectively.

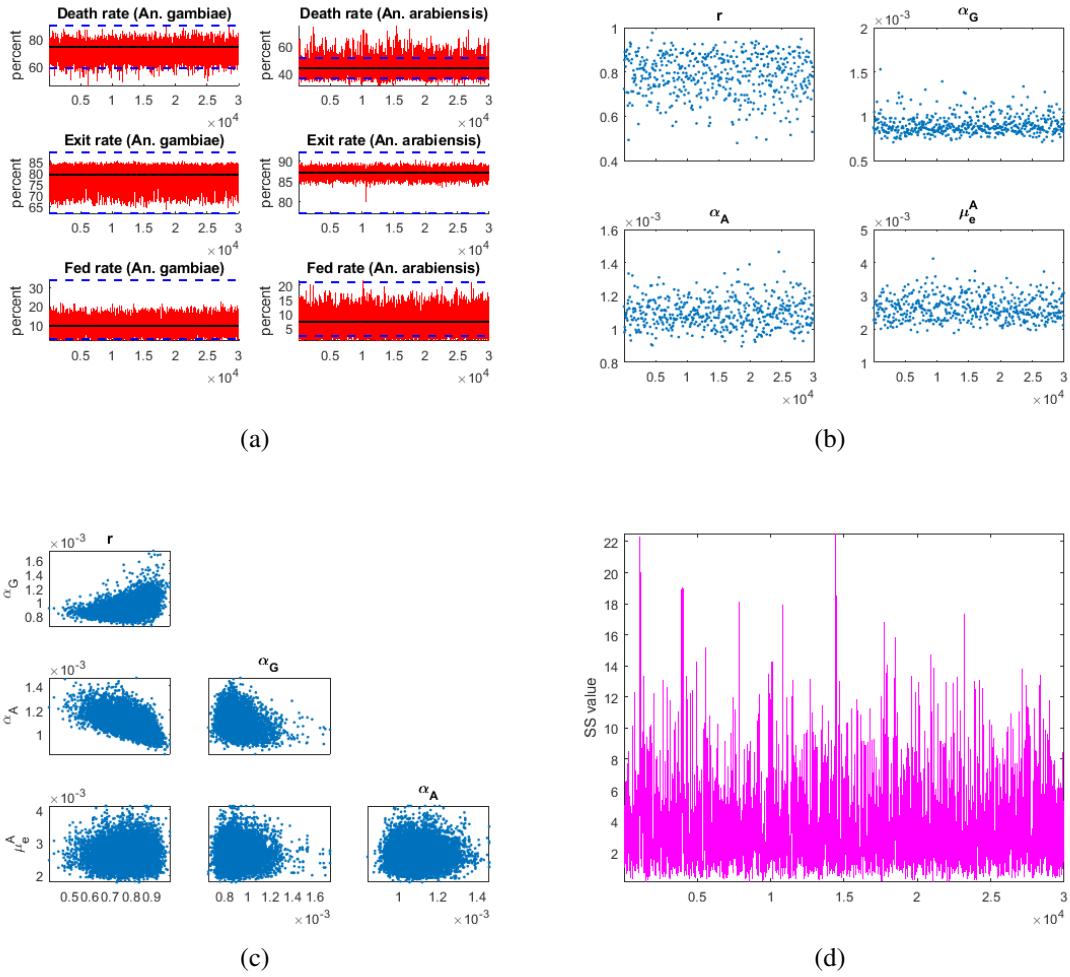


Figure 16. Results of model calibration with the data for LLIN treated with IconMaxx LN kit. (a) Model outputs vs measured data. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

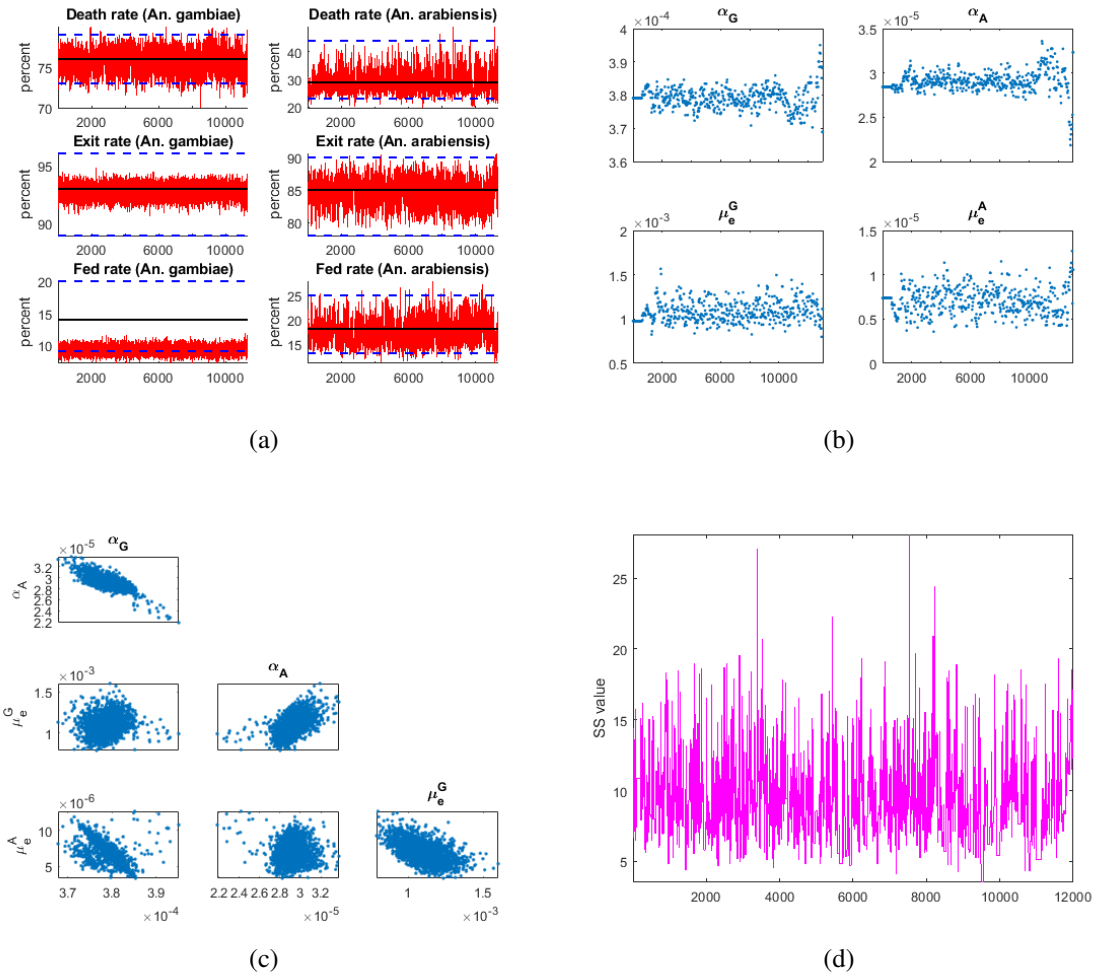


Figure 17. Results of model calibration with the data for LLIN treated with Deltamethrin kit. (a) Model outputs vs measured data. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

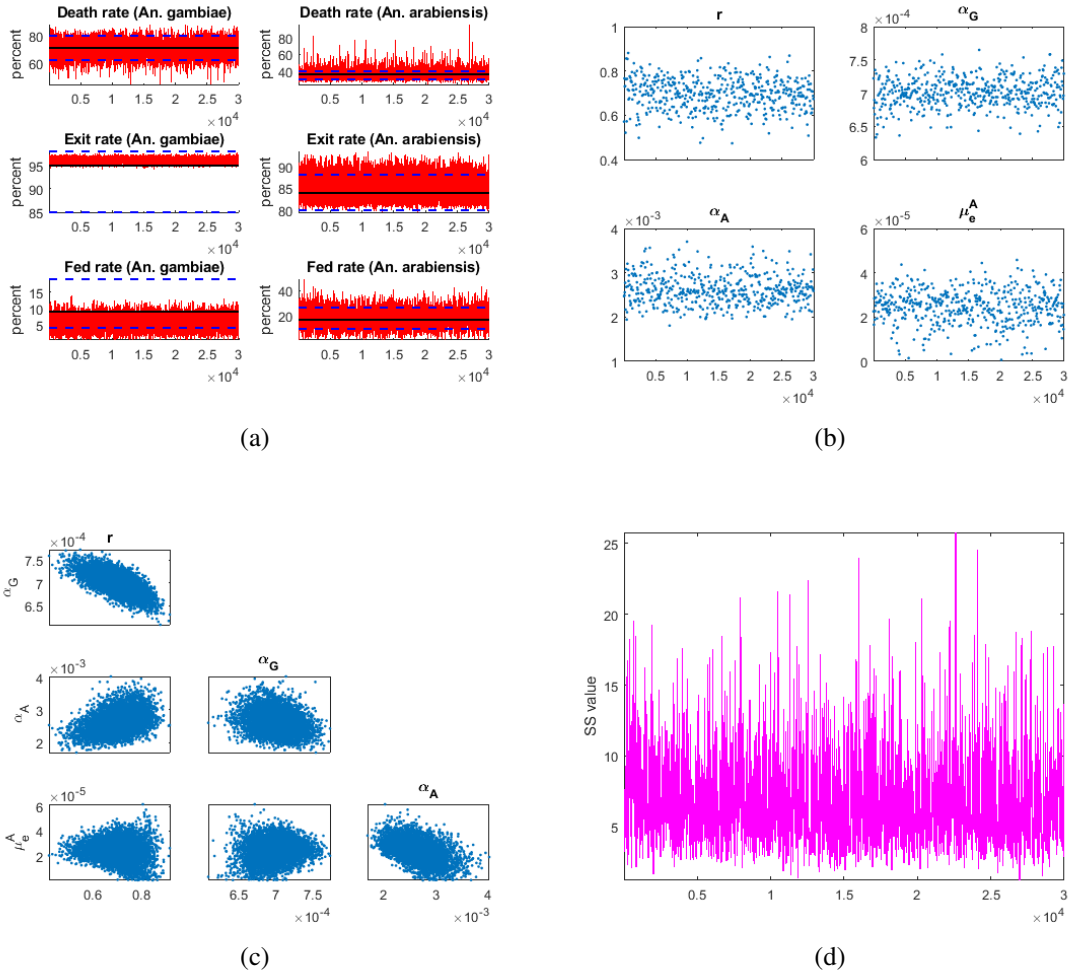


Figure 18. Results of model calibration with the data for LLIN treated with Alphacypermethrin kit. (a) Model outputs vs measured data. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

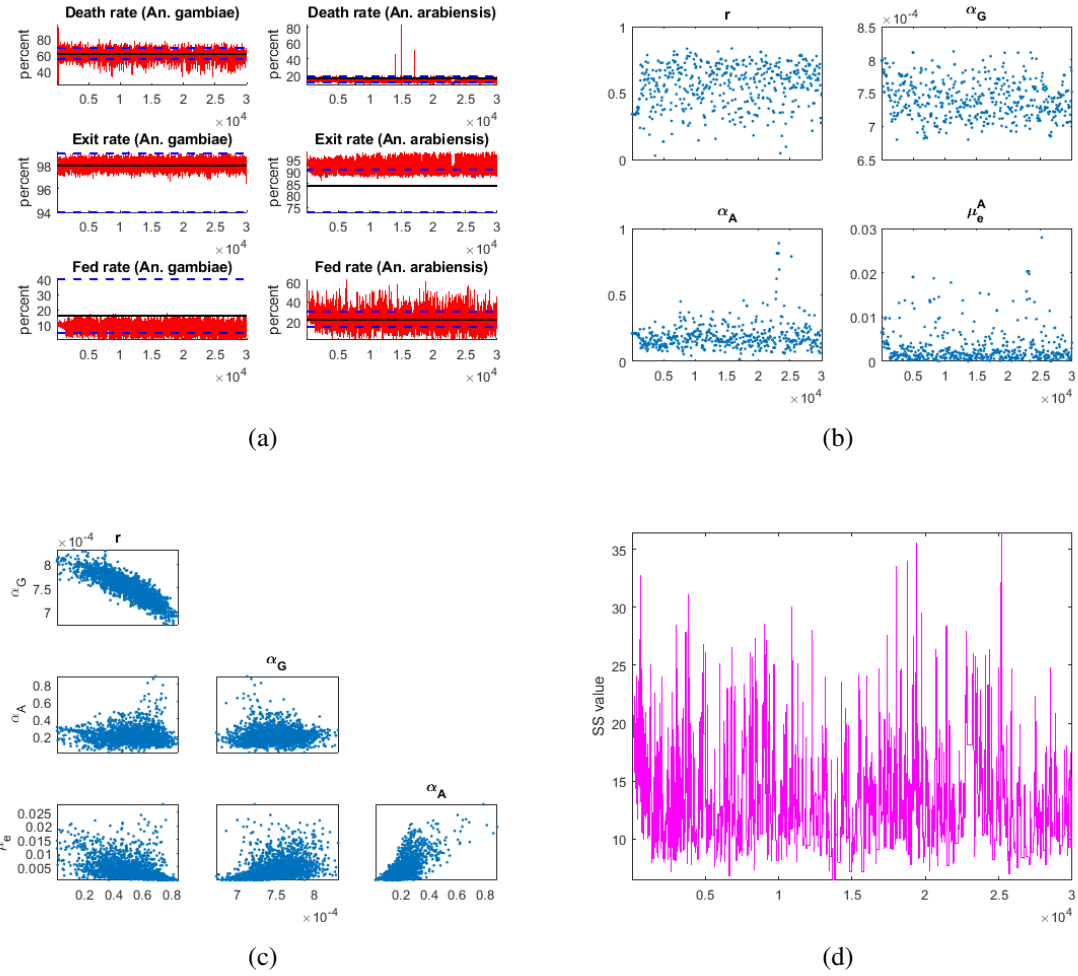


Figure 19. Results of model calibration with the data for LLIN treated with Olyset LN kit. (a) Model outputs vs measured data. (b) Parameter chains. (c) Pairwise distribution of parameters. (d) Sum of squared chain.

The mean value of the Monte Carlo samples in comparison with the real data for each chemical is presented in Table 8 for *An. gambiae* and *An. arabiensis* respectively.

Table 8. Mean value of Monte Carlo samples alognside data from experimental hut trials with bednets treated with several insecticides.

An. gambiae								
Chemicals / Respnse	**Simulated Outputs' Mean Value**				**Actual Data Value/Range**			
	IconMaxx	Deltamethrin	Alphacypermethrin	Olyset	IconMaxx	Deltamethrin	Alphacypermethrin	Olyset
Percentage mortality	70.40	75.62	70.25	60.09	74(59-89)	76(73-79)	71(62-80)	61(54-69)
Percentage exit rate	79.27	92.74	96.11	98.10	79(62-89)	93(89-96)	95(85-98)	98(94-99)
Percentage fed rate	9.48	9.27	5.57	7.88	9(2-34)	14(9-20)	9(4-19)	16(5-40)
An. arabiensis								
Chemicals / Respnse	**Simulated Outputs' Mean Value**				**Actual Data Value/Range**			
	IconMaxx	Deltamethrin	Alphacypermethrin	Olyset	IconMaxx	Deltamethrin	Alphacypermethrin	Olyset
Percentage mortality	43.40	28.70	35.76	12.38	44(37-51)	29(23-44)	36(31-40)	15(11-18)
Percentage exit rate	86.89	84.71	84.63	95.56	87(77-92)	85(78-90)	84(80-88)	84(73-91)
Percentage fed rate	6.77	17.33	17.94	21.55	7(2-21)	18(13-25)	17(10-27)	22(15-30)

It can be seen from the above results presented, that the model is a real good fit to the response measurements and the variability of the simulated model output has a close match with the error bars of the measurement. This is evident in Figures 16a, 17a, 18a and 19a; where the blue lines denote the error bounds of the actual measurement, depicted by the black constant line. On the other hand, all the sampled parameter values, via which the above fits of the model to the measurements are obtained, are well identified and are bounded from above and below. Thus, upon fixing the supposed variable parameters that were not accurately identified in preliminary samples, it can be seen that their exist relatively low uncertainties in terms of parameter estimation. However, the pairwise correlation plots do not reveal any consistent correlation between the sampled parameters.

Next, we provide logical explanations of the implications of the initial values which gave rise to the distribution of parameters in Figures 16b, 17b, 18b and 19b, in relation with the values of the measured factors. Considering the repellent effects which disorient the movement of mosquitoes, making it difficult for them to detect the host, we have that IconMaxx is highly repulsive and this is apparent from the low fed rate it features for both mosquito species. This is followed by Alphacypermethrin which has has a repulsive strength of approximately 0.5. Moreover, Deltamethrin and Olyset have very low repulsion intensity and this is evident in the high feeding rate associated with the bed nets treated with these chemicals. Thus, the repulsive strength for Deltamethrin is particularly considered as negligible and we fix the parameter $r = 0$ for this chemical. Fixing $r = 0$ for Deltamethrin implies that, its probability of rejection will generally be zero in the simulated experiments. The detoxification rates for all the chemicals are consistently higher in *An. arabiensis* as compared to that of *An. gambiae* of which is accounted for in the death rate of *An. gambiae* being up to twice higher than that of *An. arabiensis* from the

datasets employed. Furthermore, while not much distinct interpretations can be given for the values the exito repellency parameters assumes in relation to the exit rates, as there are other parameters that rule the exit rates, it can be noticed that the rate of increase in exito repellency is relatively lower for Deltamethrin that features a negligible repulsive strength and higher for IconMaxx which on the other hand, has a higher repulsive strength. Again, the values of exito repellency parameters are consistently higher for *An. gambiae* as compared to *An. arabiensis* since the exit rates for *An. gambiae* are relatively higher than that of *An. arabiensis* from the data presented in [22]. Moreover, preliminary samples showed that the parameter, μ_e^G is unbounded for IconMaxx, Alphacypermethrin and Olyset and as such, the said parameter was fixed to be equal to 0.8, 1 and 1 respectively for the aforementioned chemicals.

5 DISCUSSION AND CONCLUSION

In this section, a general discussion of the report, alongside the conclusions reached is provided. Recommendations based on the report findings are given and some relevant extensions of the work are finally stated.

5.1 Discussion

In the present work, we introduce a discrete agent-based model of mosquito host-seeking behavior when confronted with two control measures; untreated net and ITNs/LLINs as we employ two malaria vector species; *An gambiae* and *An. arabiensis*. First, the overnight host-seeking behavior of mosquitoes was simulated in case of untreated net where we assume no difference in the host-seeking behavior of both mosquito species. In this case, the variable parameters include probability of penetration through the net barrier and probability of exiting from the hut into the window traps. The parameters ruling mosquito attraction to the host were fixed while variable parameters were sampled from previously generated MCMC chains at each successive iteration of the algorithm. Next, we simulate the overnight trials with the treated net where a dissimilar host-seeking behavior is assumed for both mosquito species. Here, in addition to the control case features, a repulsion by net repellent and insecticide-induced mortality are taken into account. In order to achieve closer correspondence with the data sets in [22], we introduced chemical detoxification rates and exito-repellency rates differently for both species such that they account for the differences in the mortality, feeding and exit rates of *An. gambiae* and *An. arabiensis* inherent in the data. Similar to the previous case, all the variable parameters were sampled from MCMC parameter chains at each iteration of the algorithm.

We identify an optimal configuration for model runs to compromise between the noisiness of the model outputs and CPU time required for computations. As a measure of statistical reliability, we use the confidence intervals reported in the paper by [22] for experimental data, where the nets were treated with the treatment kits (IconMaxx LN). By experimental configuration, we study a combination of number of repetitions and the size of mosquito population over 20000 Monte Carlo iterations. A good combination of the minimum population size and number of repeated simulations was identified to be 600 mosquitoes with 4 repeated simulations. Furthermore, we studied how changes in the variable parameter values of the control case model affects the model outputs as we considered two factors; percentage exit rate and percentage fed rate. It was affirmed that p_{net}

is the principal parameter that rules the feeding rate whereas p_{hut} basically determines the value that the percentage exit rate assume. Thus we have that, increasing p_{net} causes a decrease in te percentage fed mosquitoes and slightly reduces the exit rate while an increase in p_{hut} brings about an increase in the percentage exit rate with a slight decrease in the percentage fed rate.

Furthermore, we calibrate the control case and treatment case models against several real data on experimental hut trials reported in certain literatures [22], [52], [53], [54]. The models are calibrated for each dataset by using Adaptive MCMC with chain length of 20,000 with each sample consisting of an average of 4 repetitions with a swarm of 600 mosquitoes. The control model really fits all the data sets considered well, within the reported experimental error bounds and the parameters well accurately identified by the datasets. In the treatment case, however, it was shown that the model fits virtually all the measurements real good and the variability of the simulated outputs has a close match with the report on data noisiness given for the datasets. While the datasets are able to correctly identify the sampled parameters and limits of the parameter values, they however, do not reveal any logical correlations between the parameters.

5.2 Conclusion

Based on the results obtained, we conclude that the fits obtained in the control case are ideal and the model is suitable for calibration with real data. On the other hand, the treatment case model has demonstrated its capabilities to replicate real experimental hut trials' data, execpt for one of the measured responses (exit rate for *An. arabiensis*) for Olyset chemical. Overall, the model calibrations evaluate the overall impact of LLINs, different for *An. gambiae* and *An. arabiensis*.

5.3 Recommendations and Future work

5.3.1 Recommendations

Based on the results of this work, we recommend the following:

- People living in places where they are predisposed to bites from Anopheline mosquitoes should always sleep under LLINs.

- Research institutions should carry out researches aimed at improving the efficacy of long-lasting insecticidal net in malaria control.

- Governments should incorporate in their budgets, the cost of making ITNs/LLINs continually available and sufficient for the households. They can also create and sponsor awareness programmes to encourage households who have the LLINs to make use of them because reports have shown that some households who have LLINs do not sleep under them [57].

5.3.2 Limitations of Study and Future Work

This work focused only on the hut experiment considering *An. gambiae* and *An. arabiensis* as the vector specie of study and ITN/LLIN alongside untreated nets, as protective measures. To that end, many possible and realizable extensions of the present work exist. These include:

- Conducting a village scale experiment with various household sizes under different degrees of protection with LLIN and afterwards, make statistical regression models to ascertain the impact of household size and degree of net coverage on malaria transmission.

- Improvising better parameterizations for modeling the host-seeking behaviors of mosquitoes in the presence of ITNs/LLINs.

- Making a comparative study of the efficacy of LLIN in controlling several anopheline mosquitoes in a hut experiment and subsequently incorporating more intervention strategies other than just the ones employed in this work so as to handle the endogenous behavioural plasticity of some mosquito species.

REFERENCES

[1] J. Sachs and P. Malaney. The economic and social burden of malaria. *Nature*, 415(6872):680, 2002.

[2] R. W. Steketee, B. L. Nahlen, M. E. Parise, and C. Menendez. The burden of malaria in pregnancy in malaria-endemic areas. *The American journal of tropical medicine and hygiene*, 64(13):28–35, 2001.

[3] G. F. Killeen, T. A. Smith, H. M. Ferguson, H. Mshinda, S. Abdulla, C. Lengeler, and S. P. Kachur. Preventing childhood malaria in africa by protecting adults from moquitoes wih insecticide-treated nets. *PLoS Medicine*, 4(7):1246–1258, 2007.

[4] S. Mandal, R. Sarkar, and S. Sinha. Mathematical models of malaria - A review. *Malaria Journal*, 10:202, 2011.

[5] D. S. Pouniotis, O. Proudfoot, G. Minigo, J. L. Hanley, and M. Plebanski. Malaria parasite interactions with the human host. *Journal of Postgraduate Medicine*, 50(1):4–30, 2004.

[6] E. Y. Klein. Antimalarial drug resistance: A review of the biology and strategies to delay emergence and spread. *Malaria journal*, 10(1):207, 2013.

[7] M. A. Braks, S. A. Juliano, and L. P. Lounibos. Superior reproductive success on human blood without sugar is not limited to highly anthropophilic mosquito species. *Medical and Veterinary Entomology*, 20(1):53–59, 2006.

[8] W. Takken and N. O. Verhulst. *Annual Review of Entomology*, 10:33–45.

[9] K. P. Paaijmans, S. S. Imbahale, M. B. Thomas, and W. Takken. The influence of mosquito resting behaviour and associated microclimate for malaria risk. *Malaria journal*, 9(1):196, 2010.

[10] WHO. Malaria entomology and vector control. *World Health Organization*, 2013.

[11] T. Dekker, M. Geier, and R. T. Carde. Carbon dioxide instantly sensitizes female yellow fever mosquitoes to human skin odours. *Journal of Experimental Biology*, 208(15):2963–2972, 2005.

[12] M. J. Lehane. *The biology of blood-sucking in insects*. 2005.

[13] R. T. Carde. Odour plumes and odour-mediated flight in insects. *Ciba Found Symp*, 200:54–70, 1996.

[14] S. Gerstl, S. Dunkley, A. Mukhtar, P. Maes, M. De Smet, S. Baker, and J. Maikere. Long-lasting insecticide-treated net usage in eastern Sierra Leone - The success of free distribution. *Tropical Medicine and International Health*, 15(4):480–488, 2010.

[15] N. J. Govella, F. O. Okumu, and G. F. Killeen. Short report: Insecticide-treated nets can reduce malaria transmission by mosquitoes which feed outdoors. *American Journal of Tropical Medicine and Hygiene*, 82(3):415–419, 2010.

[16] J. A. Rozendaal. *Vector control: methods for use by individuals and communities*. World Health Organization, 1997.

[17] G. Tesfaye, B. Yemane, and W. Alemayehu. Low long-lasting insecticide nets (LLINs) use among household members for protection against mosquito bite in kersa, Eastern Ethiopia. *Tropical Medicine and International Health*, 15(4):480–488, 2012.

[18] A. Bomblies. Agent-based modeling of malaria vectors: the importance of spatial simulation. *Parasites & Vectors*, (1):308, 2014.

[19] W. Takken, L. J. Van, and W. Adam. Inhibition of host-seeking response and olfactory responsiveness in anopheles gambiae following blood feeding. *Journal of Insect Physiology*, 47(3):303–310, 2001.

[20] M. R. Reddy, H. J. Overgaard, S. Abaga, V. P. Reddy, A. Caccone, A. E. Kiszewski, and M. A. Slotman. Outdoor host seeking behaviour of anopheles gambiae mosquitoes following initiation of malaria vector control on bioko island, equatorial guinea. *Malaria Journal*, 10(1):184, 2011.

[21] S. Majeed. *Odour-mediated host preference in mosquitoes*. Wiley New York, 2013.

[22] J. Kitau, R. M. Oxborough, P. K. Tungu, J. Matowo, R. C. Malima, S. M. Magesa, J. Bruce, F. W. Mosha, and M. W. Rowland. Species shifts in the anopheles gambiae complex: do llins successfully control anopheles arabiensis? *PLoS one*, 7(3):e31481, 2012.

[23] A. Shcherbacheva, H. Haario, and G. F. Killeen. Modelling host-seeking behavior of african malaria vector mosquitoes in presence of long-lasting insecticidal nets.

[24] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*. 2004.

[25] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state by fast computing machines. *The Journal of Chemical Physics*, 21(1953):1087–1092, 1953.

[26] W. K. Hastings. Monte carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[27] I. S. Mbalawata. Adaptive markov chain monte carlo and bayesian filtering for state space models. *Acta Universitatis Lappeenrantaensis*, 2014.

[28] H. Haario, E. Saksman, and J. Tamminen. An adaptive metropolis algorithm. *Bernoulli*, 7(2):223, 2001.

[29] H. Haario, M. Laine, A. Mira, and E. Saksman. DRAM: Efficient adaptive MCMC. *Statistics and Computing*, 16(4):339–354, 2006.

[30] A. Mira. On Metropolis-Hastings algorithms with delayed rejection. *Metron*, 59(4):231–241, 2001.

[31] L. Tierney and A. Mira. Some adaptive monte carlo methods for bayesian inference. *Statistics in Medicine*, 18(17-18):2507–15, 1999.

[32] S. Sinharay. Assessing convergence of the markov chain monte carlo algorithms: A review. *ETS Research Report Series*, 21(6):1–37, 2003.

[33] B. Smith. Bayesian output analysis program (BOA). *Journal of Statistical Software*, 21(11):1–37, 2001.

[34] M. K. Cowles and B. P. Carlin. Markov chain monte carlo convergence diagnostics: a comparative review. *Journal of the American Statistical Association*, 91(434):883–904, 1996.

[35] A. Gelman and D. B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–511, 1992.

[36] W. F. Snow. Field estimates of the flight speed of some West African mosquitoes. *Ann Trop Med Parasitol*, 74(2):239–242, 1980.

[37] J. Crank. *The mathematics of diffusion*. Oxford university press, 1979.

[38] J. Lafferty and G. Lebanon. Information diffusion kernels. *Advances in Neural Information Processing Systems*, 7:391–398, 2003.

[39] D. Butler. Malaria. *Nature*, 74:284, 2013.

[40] L. J. Zwiebel and W. Takken. Olfactory regulation of mosquito-host interactions. *Journal of Insect Physiology*, 34(7):645–652, 2004.

[41] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[42] J. Spitzen, C. W. Spoor, F. Grieco, C. Braak, J. Beeuwkes, S. P. Brugge, S. Kranenbarg, L. P. Noldus, J. L. Leeuwen, and W. Takken. A 3D Analysis of flight behavior of anopheles gambiae sensu stricto malaria mosquitoes in response to human odor and heat. *PLoS ONE*, 8(5), 2013.

[43] F. O. Okumu, E. Mbeyela, G. Lingamba, J. Moore, A. J. Ntamatungiro, D. R. Kavishe, M. G. Kenward, E. Turner, L. M. Lorenz, and S. J. Moore. Comparative field evaluation of combinations of long-lasting insecticide treated nets and indoor residual spraying, relative to either method alone, for malaria prevention in an area where the main vector is anopheles arabiensis. *Parasites & vectors*, 6(1):46, 2013.

[44] A. N. Clements and G. D. Paterson. The analysis of mortality and survival rates in wild populations of mosquitoes. *Journal of Applied Ecology*, 18(2):373–399, 1981.

[45] M. T. Gillies and T. J. Wilkes. A comparison of the range of attraction of animal baits and of carbon dioxide for some West African mosquitoes. *Bulletin of Entomological Research*, 59(3):441–456, 1969.

[46] WHO. Guidelines for testing mosquito adulticides for indoor residual spraying and treatment of mosquito nets: Technical report. *WHO*, 2006.

[47] M. T. Gillies. A comparison of the range of attraction of animal baits and of carbon dioxide for some West African mosquitoes. *Bulletin of Entomological Research*,

48(3):333–559, 1957.

[48] G. F. Killeen, N. Chitnis, S. J. Moore, and F. O. Okumu. Target product profile choices for intra-domiciliary malaria vector control pesticide products: repel or kill? *Malaria Journal*, 10(1):207, 2011.

[49] T. J. Sullivan. *Introduction to uncertainty quantification*. Springer, 2015.

[50] A. Saltelli, K. Chan, and E. M. Scott. *Sensitivity analysis*. Wiley New York, 2000.

[51] ISPOR 17th Annual International Meeting. *Fundamentals of model calibration: Theory and Practice*, volume 17. OPTUMInsight Washington, DC USA, 2012.

[52] R. C. Malima, R. M. Oxborough, P. K. Tungu, C. Maxwell, I. Lyimo, V. Mwingira, F. W. Mosha, J. Matowo, S. M. Magesa, and M. W. Rowland. Behavioural and insecticidal effects of organophosphate-, carbamate-and pyrethroid-treated mosquito nets against african malaria vectors. *Medical and Veterinary Entomology*, 23(4):317–325, 2009.

[53] R. C. Malima, S. M. Magesa, P. K. Tungu, V. Mwingira, F. S. Magogo, W. Sudi, F. W. Mosha, C. F. Curtis, C. Maxwell, and M. Rowland. An experimental hut evaluation of olyset® nets against anopheline mosquitoes after seven years use in tanzanian villages. *Malaria Journal*, 7(1):38, 2008.

[54] F. W. Mosha, I. N. Lyimo, R. M. Oxborough, J. Matowo, R. Malima, E. Feston, R. Mndeme, F. Tenu, M. Kulkarni, and C. A. Maxwell. Comparative efficacies of permethrin-, deltamethrin-and α-cypermethrin-treated nets, against anopheles arabiensis and culex quinquefasciatus in northern tanzania. *Annals of Tropical Medicine & Parasitology*, 102(4):367–376, 2008.

[55] G. A. Kerkut and L. I. Gilbert. *Comprehensive insect physiology, biochemistry and pharmacology*. Pergamon Press, New York, 1985.

[56] S. B. Ogoma, L. M. Lorenz, H. Ngonyani, R. Sangusangu, M. Kitumbukile, M. Kilalangongono, E. T. Simfukwe, A. Mseka, E. Mbeyela, and D. Roman. An experimental hut study to quantify the effect of ddt and airborne pyrethroids on entomological parameters of malaria transmission. *Malaria Journal*, 13(1):131, 2014.

[57] The international Federation's Global Agenda. Long-lasting, insecticide-treated nets

(LLIN) scale-up and hang-up programme. *Bulletin of the World Health Organization*, (4):290–6, 2014.

APPENDICES

Appendix 1. Control Case Data

Table A1.1. Data for experimental hut trials in the control case.

Source	Percentage exit rate	Percentage fed rate
[22]	80	53
[52]	71(68-73)	41(38-44)
[53]	91.1(85.8-96.4)	26.8(18.6-35.0)
[54]	82.5(79.0-85.6)	24.0(20.5-27.8)
[54]	70.7(63.8 76.7)	68.6(61.7-74.8)