Supplement to "Informing policy via dynamic models: Eliminating cholera in Haiti"

Jesse Wheeler, Anna Rosengart, Zhuoxun Jiang,
Kevin Hao En Tan, Noah Treutle, Edward L. Ionides

Department of Statistics, University of Michigan

September 13, 2022

Supplementary Content

S1 Markov chain and differential equation interpretations of compartment flow rates

In Sections 2.1, 2.2 and 2.3 we define compartment models in terms of their flow rates. For a discrete population model, these rates define a Markov chain. For a continuous and deterministic model, the rates define a system of ordinary differential equations. Here, we add additional details to clarify the mapping from a collection of rate functions to a fully specified process. Our treatment follows Bretó et al. (2009).

A general compartment model is a vector-valued process $X(t) = (X_1(t), \ldots, X_c(t))$ denoting the (integer or real-valued) counts in each of c compartments. The compartments may also have names, but to set up general notation we simply refer to them by their numerical index. The basic characteristic of a compartment model is that $X(t)$ can be written in terms of the flows $N_{ij}(t)$ from i to j, together with flows into and out of each compartment denoted by $N_{\bullet i}(t)$ and $N_{i\bullet}(t)$ respectively. These flows are required to satisfy a "conservation of mass" identity:

$$X_i(t) = X_i(0) + N_{\bullet i}(t) - N_{i\bullet}(t) + \sum_{j \neq i} N_{ji}(t) - \sum_{j \neq i} N_{ij}(t). \tag{S1}$$

Each *flow* $N_{ij}(t)$ is associated with a *rate* function $\mu_{ij} = \mu_{ij}(t, X(t))$, where we include the possibility that i or j takes value $\bullet$.

There are different ways to use a collection of rate functions to build a fully specified model. We proceed to describe the ones we use in this paper: via a system of ordinary differential equations (Sec. S1.1), a simple Markov counting system (Sec. S1.2), and an overdispersed Markov counting system (Sec. S1.3). Other representations include stochastic differential equations driven by Gaussian noise or Gamma noise (Bhadra et al., 2011).

S1.1 Ordinary differential equation (ODE) interpretation

A basic deterministic specification is

$$dN_{ij}/dt = \mu_{ij}\big(t, X(t)\big) X_i(t), i \in 1\!:\!c, j \in 1\!:\!c \cup \{\bullet\}, i \neq j, \tag{S2}$$

where $\mu_{ij}\big(t, X(t)\big)$ is called a per-capita rate or a unit rate. Flows into the system require special treatment since $X_i(t)$ in (S2) is not defined for $i = \bullet$. Instead, we specify

$$dN_{\bullet i}/dt = \mu_{\bullet i}\big(t, X(t)\big). \tag{S3}$$

This is the the interpretation and implementation used for Model 2 in our study.

S1.2 Simple Markov counting system interpretation

A continuous time Markov chain can be specified via its infinitesimal transition probabilities. A basic approach to this is to define

$$\mathbb{P}\big[N_{ij}(t+\delta) - N_{ij}(t) = 0 \,|\, X(t)\big] = 1 - \delta\mu_{ij}\big(t, X(t)\big) + o(\delta), \tag{S4}$$

$$\mathbb{P}\big[N_{ij}(t+\delta) - N_{ij}(t) = 1 \,|\, X(t)\big] = \delta\mu_{ij}\big(t, X(t)\big) + o(\delta), \tag{S5}$$

for $i \in 1{:}c$ and $j \in 1{:}c \cup \{\bullet\}$ with $i \neq j$. As with the ODE case, we need special attention for flows into the system, and we define

$$\mathbb{P}\big[N_{\bullet i}(t+\delta) - N_{\bullet i}(t) = 0 \,|\, X(t)\big] = 1 - \delta\mu_{\bullet i}\big(t, X(t)\big) + o(\delta), \tag{S6}$$

$$\mathbb{P}\big[N_{\bullet i}(t+\delta) - N_{\bullet i}(t) = 1 \,|\, X(t)\big] = \delta\mu_{\bullet i}\big(t, X(t)\big) + o(\delta). \tag{S7}$$

Together with the initial conditions $X(0)$, equations (S4)–(S7) define a Markov chain. Each flow is a simple counting process, meaning a non-decreasing integer-valued process that only has jumps of size one. We therefore call the Markov chain a simple Markov counting system (SMCS). The infinitesimal mean of every flow is equal to its infinitesimal variance (Bretó and Ionides, 2011) and so an SMCS is called equidispersed. We note that the special case of Model 1 used in Lee et al. (2020a) (with $\sigma_{\text{proc}} = 0$) is an SMCS. To permit more general mean-variance relationships for a Markov counting system, we must permit jumps of size greater than one. The utility of overdispersed models, where the infinitesimal variance of the flow exceeds the infinitesimal mean, has become widely recognized (Stocks et al., 2020; He et al., 2010).

S1.3 Overdispersed Markov counting system interpretation

Including white noise in the rate function enables the possibility of an overdispersed Markov counting system (Bretó and Ionides, 2011; Bretó et al., 2009; He et al., 2010). Since rates should be non-negative, Gaussian noise is not appropriate and gamma noise is a convenient option that has found various applications (Romero-Severson et al., 2015; Subramanian et al., 2020). Specifically, we consider a model given by

$$\mu_{ij}\big(t, X(t)\big) = \bar{\mu}_{ij}\big(t, X(t)\big)\, d\Gamma_{ij}(t)/dt, \tag{S8}$$

where $\Gamma_{ij}(t)$ is a stochastic process having independent gamma distributed increments, with

$$\mathbb{E}\big[\Gamma_{ij}(t)\big] = t, \quad \mathrm{Var}\big[\Gamma_{ij}(t)\big] = \sigma_{ij}^2 t. \tag{S9}$$

Formally interpreting the meaning of (S8) is not trivial, and we do so by defining the solution of (S8) to be the limit of an Euler scheme. Therefore, the numerical scheme in Sec. S2 can be taken as a definition of the meaning of (S8). The Markov chain defined by the limit of this Euler scheme as the step size decreases is an overdispersed Markov counting system, with the possibility of instantaneus jumps of size greater than one (Bretó and Ionides, 2011).

S2 Numerical solutions to compartment models

Models may be fitted and their implications assessed via numerical solutions (i.e., simulations) from the model equations. All the analyses we consider have this simulation-based property, known as plug-and-play or equation-free or likelihood-free. The numerical solutions to the model are arguably of more direct scientific interest than the exact solutions to the postulated equations. For ODE models, numerical methods are well studied and a standard numerical solution package such as `deSolve` in `R` is adequate for our purposes. For SMCS and ODMCS models, exact schemes are feasible when the number of events is small, which may be the case for small populations. However, for applicability to larger populations, we use instead the following Euler scheme. Write δ for an Euler time step, and ΔN_{ij} for the numerical approximation to $N_{ij}(t+\delta) - N_{ij}(t)$ given $X(t)$. For

each i and j in $1:c \cup \{\bullet\}$ with $i \neq j$, we draw independent Gamma distributed noise increments with mean δ and variance $\sigma_{ij}^2\delta$, denoted using a mean-variance parameterization of the gamma distribution as

$$\Delta\Gamma_{ij} \sim \text{gamma}(\delta, \sigma_{ij}^2\delta). \tag{S10}$$

In the case of an SMCS model, $\sigma_{ij} = 0$ for all i and j, so we have $\Delta\Gamma_{ij} = \delta$. Then, for $i \neq \bullet$ and $j \neq i$, and writing

$$\mu_{ij} = \bar{\mu}_{ij}\big(t, X(t)\big)\Delta\Gamma_{ij}/\delta, \tag{S11}$$

we calculate transition probabilities

$$p_{ij} = \exp\left\{-\sum_{k\in 1:c\cup\{\bullet\}} \mu_{ik}\,\delta\right\} \frac{\mu_{ij}}{\sum_{k\in 1:c\cup\{\bullet\}} \mu_{ik}}, \tag{S12}$$

$$p_{ii} = 1 - \sum_{j\neq i} p_{ij}. \tag{S13}$$

These probabilities correspond to competing hazards for every individual in compartment i to transition to some compartment j, interpreting $j = i$ to mean that the individual remains in i. Then, $\big(\Delta N_{i1}, \ldots, \Delta N_{ic}, \Delta N_{i\bullet}\big)$ has the multinomial distribution where $X_i(t)$ individuals are allocated independently to $1 : c \cup \{\bullet\}$ with probabilities given by (S12) and (S13). We use the `reulermultinom` function in the `pomp` package to draw from this multionomial distribution.

For the case $i = \bullet$, one can use

$$\Delta N_{\bullet j} \sim \text{poisson}(\mu_{\bullet j}\delta), \tag{S14}$$

an independent Poisson random variable with mean $\mu_{\bullet j}\delta$, as was done in Model 1.

Another common approach is to balance the total number of flows in and out of the compartment, i.e., $\sum_i N_{\bullet i}(t) = \sum_i N_{i\bullet}(t)$, in order to make the model consistent with the known total population, as was done in Models 2 and 3. In this case, we formally model the death rate as a rate of returning to the susceptible class S, and use external transitions from $\bullet$ into S to describe only net population increase.

S3 Initial Values

To perform inference on POMP models, it is necessary to propose an initial density for the latent process $f_{X_0}(x_0; \theta)$. This density is used to obtain initial values of the latent state when fitting and evaluating the model. For each of the models considered in this analysis, the initial conditions are derived by enforcing the model dynamics on reported cholera cases. It is also sometimes necessary to fit some initial value parameters in order to help determine starting values for weakly identifiable compartments. In the following subsections, we mention initial value parameters that were fit for each model.

S3.1 Model 1

For this model, the number of individuals in the Recovered and Asymptomatic compartments are set to zero, but the initial proportion of Infected and Exposed individuals is estimated as initial value parameters ($I(0)$ and $E(0)$, respectively) using the MIF2 algorithm. Finally, the initial proportion of Susceptible individuals S_0 is calculated as $S(0) = 1 - I(0) - E(0)$.

S3.2 Model 2

Model 2 assumes that the initial vales are constant, and no parameters need to be estimated. Starting value for latent state compartments are chosen so as to enforce the model dynamics on the observed number of cases. Specifically, the model sets $I_u(0) = y^*_u(0)/\rho$ for each unit $u \in 1:10$, where $I_u(t)$ is the number of infected individuals in unit u at time t, $y^*_u(t)$ is the reported number of cases, and ρ is the reporting rate.

S3.3 Model 3

We use the reported cases at the start of the pandemic to approximate the number of Asymptomatic, Infectious, and Recovered individuals in each department $u \in 1:U$ using the same approximation as provided in Eq. 34. The susceptible compartment is initialized so that the sum $S_u(0) + I_u(0) + A_u(0) + \sum_k R_{u,k}(0) = \text{population}_u$. The bacteria compartment is then initialized using Eq. (S15):

$$B_u(0) = \left[1 + a\big(\xi_u\big)^r\right] D_i \, \mu_W \left[I_u(0) + \epsilon_W A_u(0)\right] \tag{S15}$$

Where $\xi_u \in (0,1)$ are initial value parameters that we introduce in order to allow some flexibility in determining the initial state of the bacteria compartment.

S4 Measurement Models

Each POMP requires specification of a measurement model, which is a statistical description of how observations on the system are obtained. In general, we used the same measurement models that were reported in Lee et al. (2020a).

S4.1 Model 1

In this model, the advantage afforded by vaccination is an increased probability that an infection is asymptomatic. Therefore, under the assumptions of this model, all reported cases are assumed to be a fraction of individuals that transition from the exposed to the infected compartment, as noted in Eq. (S16).

$$y_t \mid \Delta N_{E_z I_z}(t) = z(t) \sim \text{NB}\left(\rho z_(t), \psi\right) \tag{S16}$$

Where y_t is the reported cholera cases at time $t \in 1:N$ and $\Delta N_{E_z I_z}(t)$ is the number of individuals who moved from compartment E_z to I_z since observation $t-1 \in 0:N-1$.

S4.2 Model 2

As mentioned in the main text, Model 2 was fit using log-transformed observed cases. We fit Model 2 using the subplex algorithm in the `subplex` package, using a Gaussian measurement model (Eq. (S17)) on the log-transformed cases as the loss function.

$$\log\left(y_{u,t} + 1\right) \mid \boldsymbol{X}^{(2)}(t) = \boldsymbol{x}^{(2)}(t) \sim \text{N}\left(\log\left(\rho \sum_z I_{u,z}(t) + 1\right), \psi^2\right) \tag{S17}$$

Where $\sum_z I_{u,z}(t)$ is the total number of individuals who are infected at time t in unit $u \in 1:U$, for all vaccination campaigns z. We note that fitting a model with this loss function is equivalent to fitting using least squares, with $\log(y_{u,t}+1)$ as the response variable.

S4.3 Model 3

In this model, reported cholera cases are assumed to stem from individuals who develop symptoms and seek healthcare. Therefore reported cases are assumed to come from an over-dispersed negative binomial model, given the increase in infected individuals:

$$y_{u,t} \mid \Delta N_{S_{uz}I_{uz}}(t) = z_u(t) \sim \mathrm{NB}\left(\rho z_u(t), \psi\right) \tag{S18}$$

Where $\Delta N_{S_{uz}I_{uz}}(t)$ is the number of individuals who moved from compartment S_{uz} to I_{uz} since observation $t-1 \in 0:N-1$.

This measurement model is a minor change from that used in Lee et al. (2020a), which allowed for a change in the reporting rate on January 1st, 2018. The fitted values of the reporting rate—before and after January 2018—were 0.97 and 0.097, respectively. This major change in reporting rate alone could explain why Model 3 in Lee et al. (2020a) failed to predict the eradication of cholera. An overnight change from near perfect to almost non-existent reporting forces the model to explain the observed reduction in reported cases as a decrease in the reporting of cases, rather than a decrease in the prevalence of cholera. This shift was justified by a "change of the case definition that occurred on January 1st, 2018"; this claim was not cited, and we could find no evidence that such a drastic change in the reporting rate would be warranted. We therefore do not allow a change in reporting rate when fitting Model 3.

S5 Block Panel Iterated Filter

The panel iterated filter (PIF) (Bretó et al., 2020) has been successfully used to fit parameters of panelPOMP models in several studies (Domeyer et al., 2022; Wale et al., 2019; Bretó et al., 2020). Like other iterated filtering algorithms, PIF iteratively combines particle filter calculations with parameter perturbations. Theoretical results show that, as the perturbations are reduced over successive iterations, the perturbed parameters converge to the MLE.

In the case of a panelPOMP, the parameter vector θ may be composed of a set of shared parameters ϕ, and a set of unit-specific parameters ψ_u, for units $u \in 1:U$. Here, we say that a parameter ψ_u for $u \in 1:U$ is unit-specific if it is not involved in the one-step transition density or the measurement model of any other unit $k \neq u \in 1:U$, and shared parameters are those that are not unit-specific. The PIF algorithm treats the parameter vector $\theta = (\phi, \psi_{1:U})$ as a complete entity and gives no distinction between shared and unit-specific parameters. Even though the random walk standard deviation is set to zero for unit-specific parameters when considering data from other units, this lack of distinction may result in particle depletion for unit-specific parameters as the number of units U grows large. This is because it is likely that, when updating the parameters for a given unit, several parameter vector θ particles will not be resampled due to having low weights. When the number of units is large, this will lead to a decreasing number of ancestral particles to sample unit-specific parameters ψ_k as k increases.

As a solution to this issue, we propose the block panel iterated filter (BPIF). In this algorithm, shared ϕ and unit-specific parameters ψ_u are handled differently when updating the parameter vector $\theta = (\phi, \psi_{1:U})$. The BPIF updates θ by updating the shared vector ϕ in the same way that is done with PIF, but index resampling for unit-specific parameters $\psi_{1:U}$ only occurs when data from the corresponding unit are being considered. In the case where there are many units, we believe that this modification will lead to less particle depletion. Pseudo-code for this algorithm is given

in Algorithm S1. We note that no theoretical properties of this algorithm have yet been derived, but early empirical results suggest that this algorithm is a an improvement to the PIF algorithm.

Algorithm S1: BPIF.
Inputs:
Simulator of initial density, $f_{X_{u,0}}(x_{u,0};\theta)$ for u in $1{:}U$.
Simulator of transition density, $f_{X_{u,n}|X_{u,n-1}}(x_{u,n}|x_{u,n-1};\theta)$ for u in $1{:}U$, n in $1{:}N_u$.
Evaluator of measurement density, $f_{Y_{u,n}|X_{u,n}}(y_{u,n}|x_{u,n};\theta)$ for u in $1{:}U$, n in $1{:}N_u$.
Data $y^*_{u,n}$,for u in $1{:}U$, n in $1{:}N_u$.
Number of iterations, M.
Number of particles, J.
Starting parameter swarm, $\Theta^0_j = \left(\Phi^0_j, \Psi^0_{1:U,j}\right)$ for $j \in 1{:}J$, $u \in 1{:}U$.
Simulator of perturbation density, $h_{u,n}(\theta|\phi;\sigma)$ for $u \in 1{:}U$, $n \in 0:N_u$.
Perturbation sequence, σ_m for $m \in 1:M$.
Output:
Final parameter swarm, $\Theta^m_j = \left(\Phi^m_j, \Psi^m_{1U,j}\right)$ for $j \in 1{:}J$, $u \in 1{:}U$.

for $m \in 1:M$ **:**
  Set $\Phi^m_{0,j} = \Phi^{m-1}_j$ for $j \in 1{:}J$;
  for $u \in 1{:}U$ **:**
    Set $\Phi^{F,m}_{u,0,j} \sim h_{u,0}(\phi|\Phi^m_{0,j};\sigma_m)$ for $j \in 1{:}J$;
    Set $\Psi^{F,m}_{u,0,j} \sim h_{u,0}(\psi|\Psi^{m-1}_{u,j};\sigma_m)$ for $j \in 1{:}J$;
    Set $\Theta^{F,m}_{u,0,j} = \left(\Phi^{F,m}_{u,0,j}, \Psi^{F,m}_{u,0,j}\right)$ for $j \in 1{:}J$;
    Initialize $X^{F,m}_{u,0,j} \sim f_{X_{u,0}}(x_{u,0};\Theta^{F,m}_{u,0,j})$ for $j \in 1{:}J$;
    for $n \in 1{:}N_u$ **:**
      Set $\Theta^{P,m}_{u,n,j} \sim h_{u,n}(\theta|\Theta^{F,m}_{u,n-1,j},\sigma_m)$ for $j \in 1{:}J$;
      $X^{P,m}_{u,n,j} \sim f_{X_{u,n}|X_{u,n-1}}(x_{u,n}|X^{F,m}_{u,n-1,j};\Theta^{P,m}_{u,n,j})$ for $j \in 1{:}J$;
      $w^m_{u,n,j} = f_{Y_{u,n}|X_{u,n}}(y^*_{u,n}|X^{P,m}_{u,n,j})$ for $j \in 1{:}J$;
      Draw $k_{1:j}$ with $P(k_j = i) = w^m_{u,n,i} / \sum_{v=1}^J w^m_{u,n,v}$ for $i,j \in 1{:}J$;
      Set $\Theta^{F,m}_{u,n,j} = \Theta^{P,m}_{u,n,k_j} = \left(\Phi^{P,m}_{u,n,k_j}, \Psi^{P,m}_{u,n,k_j}\right)$ and $X^{F,m}_{u,n,j} = X^{P,m}_{u,n,k_j}$ for $j \in 1{:}J$;
    end
    Set $\Theta^m_{u,j} = \left(\Phi^{F,m}_{u,N_u,j}, \Psi^{F,m}_{u,N_u,j}\right)$ for $j \in 1{:}J$;
  end
  Set $\Theta^m_j = \left(\Phi^m_{U,j}, \Psi^m_{1:U,j}\right)$ for $j \in 1{:}J$;
end

S6 Translating to Lee et al. (2020a) notation

Since the models in Lee et al. (2020a) were developed independently, the choice of notation varies inconsistently between models. For our reanalysis, we rename parameters to provide a unified notation facilitating comparison between models. Table S-1 maps this notation back to the original notations, for reference.

Parameter	Our Notation	Lee et al. (2020a)		
		1	2	3
Reporting Rate	ρ	ρ	ρ	ϵ_1, ϵ_2
Mixing Coefficient	ν	ν	—	—
Measurement Over-Dispersion	ψ	τ	—	p
Birth Rate	μ_S	μ	—	—
Natural Mortality Rate	δ	δ	—	μ
Cholera Mortality Rate	δ_C	—	—	α
Latent Period	$1/\mu_{EI}$	$1/\sigma$	$1/\gamma_E$	—
Recovery Rate	μ_{IR}	γ	γ	γ
Loss of Immunity	μ_{RS}	α	σ	ρ
Symptomatic Ratio	f	$1-\theta_0$	k	σ
Asymptomatic Relative Infectiousness	ϵ	κ	red_β	—
Human-to-Water Shedding	μ_W	—	μ	θ_I
Asymptomatic Relative Shedding	ϵ_W	—	red_μ	θ_A/θ_I
Seasonal Amplitude	a	—	α_s	λ
Transmission	β	β	β	c
Water-to-Human	β_W	—	β_W	β
Bacteria Mortality	δ_W	—	δ	μ_β
Vaccination Efficacy	θ	θ_{vk}	$\theta_1, \theta_2, \theta_{1_5}, \theta_{2_5}$	η_{1d}, η_{2d}
Process Over-dispersion	σ_{proc}	—	—	σ_w

Table S-1: Translations between our common notation and notation used in Lee et al. (2020a)

S7 Lee et al. (2020) Replication

In this article we claimed that we were able to obtain better fits to the observed data using the same models that were proposed in Lee et al. (2020a). Along with visual comparisons to the data, this claim was supported by comparing likelihoods and AIC values in Table 1. Because model likelihoods were not provided in Lee et al. (2020a), it is necessary to replicate these models in order to obtain likelihood estimates. Here we would like to thank the authors of Lee et al. (2020a), who provided detailed descriptions of their models, which enabled us to build on their work. In the following subsections, we use our `R` package `haitipkg` to reproduce some of the results of Lee et al. (2020a). This reproduction allows us to estimate the likelihoods of the Lee et al. (2020a) version of Models 1–3, and also provides a demonstration on reproducibility.

S7.1 Model 1 Replication

The model was implemented by a team at Johns Hopkins Bloomberg School of Public Health (hereafter referred to as the Model 1 authors) in the `R` programming language using the `pomp` package (King et al., 2016). Original source code is publicly available with DOI: 10.5281/zenodo.3360991. Despite having source code available, no point estimates for model parameters were provided in Lee et al. (2020a). According to the supplement materials, this was because model realizations from a single parameter set retained substantial variability, but multiple realizations from a collection of parameter sets resulted in a reasonable visual fit to the data. We are also inclined to believe that the use of multiple parameter values was in part intended to account for parameter uncertainty (as mentioned in our main text), an effort by the Model 1 authors that we applaud. We note, however, that simulations from each of the parameter sets were treated with equal importance when being used to diagnose the model fit and make inference on the system. This practice is problematic given Figures S8 and S9 of the supplement material, which suggest that some parameter sets that were used for inference were several hundred log-likelihood units lower than the best performing sets of parameters. When accounting for parameter uncertainty, one should instead weigh resulting model projections based on the likelihood of the set of parameters that were used to obtain the model realizations, as was done in King et al. (2015).

Not providing a point estimate for model parameters also has the effect of making model reproduction a more difficult task. We therefore rely on the range and median values for each of the final model parameters are provided in Tables S10 and S11 of the supplement, and bivariate relationships between some of the fitted parameters (Figures S8 and S9) in order to replicate their results.

We begin by constructing 300 different parameter sets based on the basic summaries and figures provided by the Model 1 authors. This was primarily done using truncated-bivariate normal distributions, where samples that are drawn outside of the desired range of parameters are re-sampled from the distribution. Because the likelihood surface is complex and almost certainly includes higher than two-level interactions between parameter values, we then perform iterated filtering for the epidemic phase of the data using the 300 different parameter sets as starting points.

Following the model fitting scheme used by the Model 1 authors, we then use these resulting parameters as starting values for the endemic phase, and refit parameters using MIF2. To do this, we use the same number of particles and MIF iterations that were used in Lee et al. (2020a). The resulting bivariate relations between ρ, τ, β_1 and ν are given in Figure S-2.

Now that we have fit the parameters, we can simulate from the models. The results of these

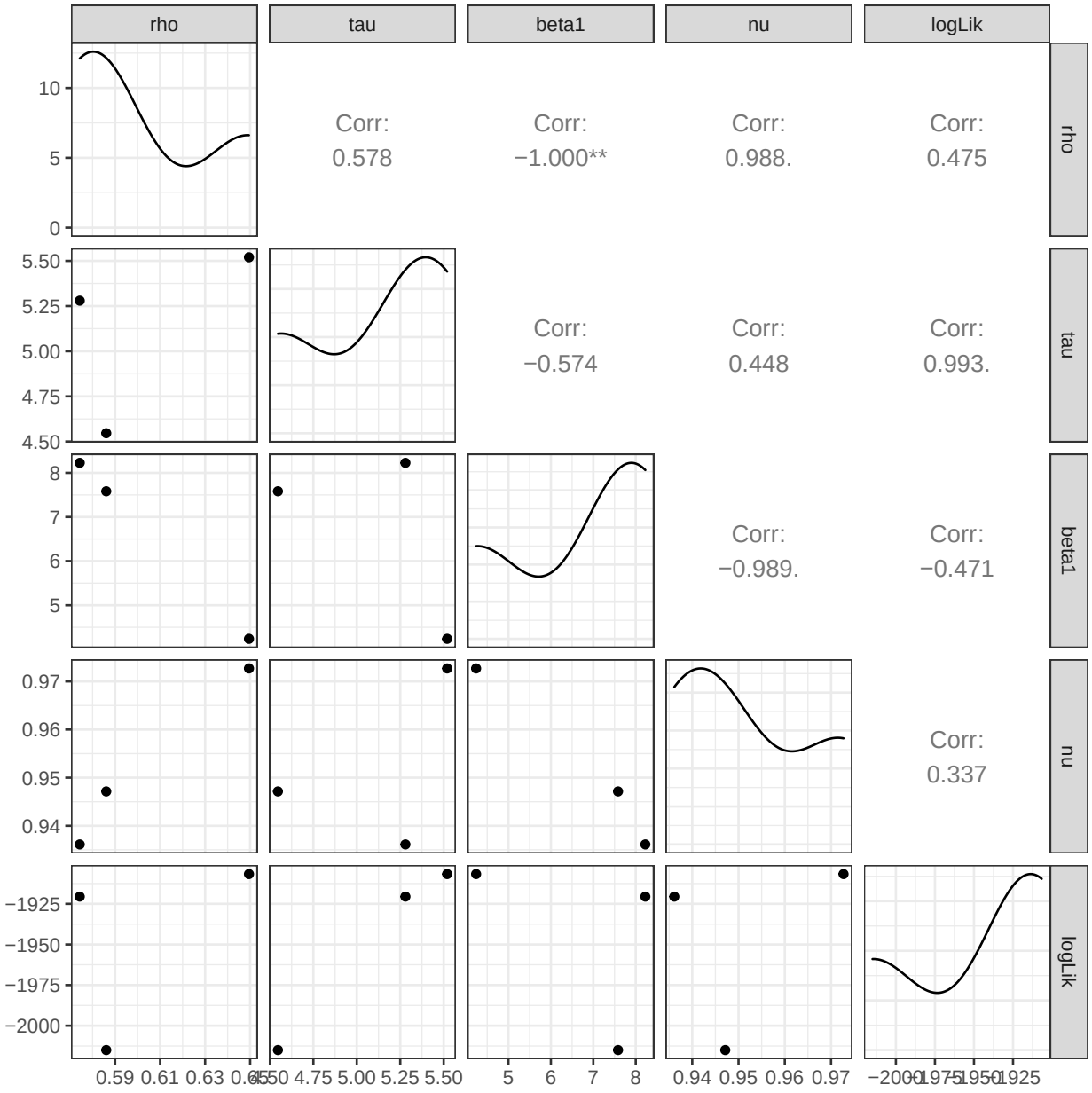

Figure S-1: Compare to Figure S8 of Lee et al. (2020) supplement.

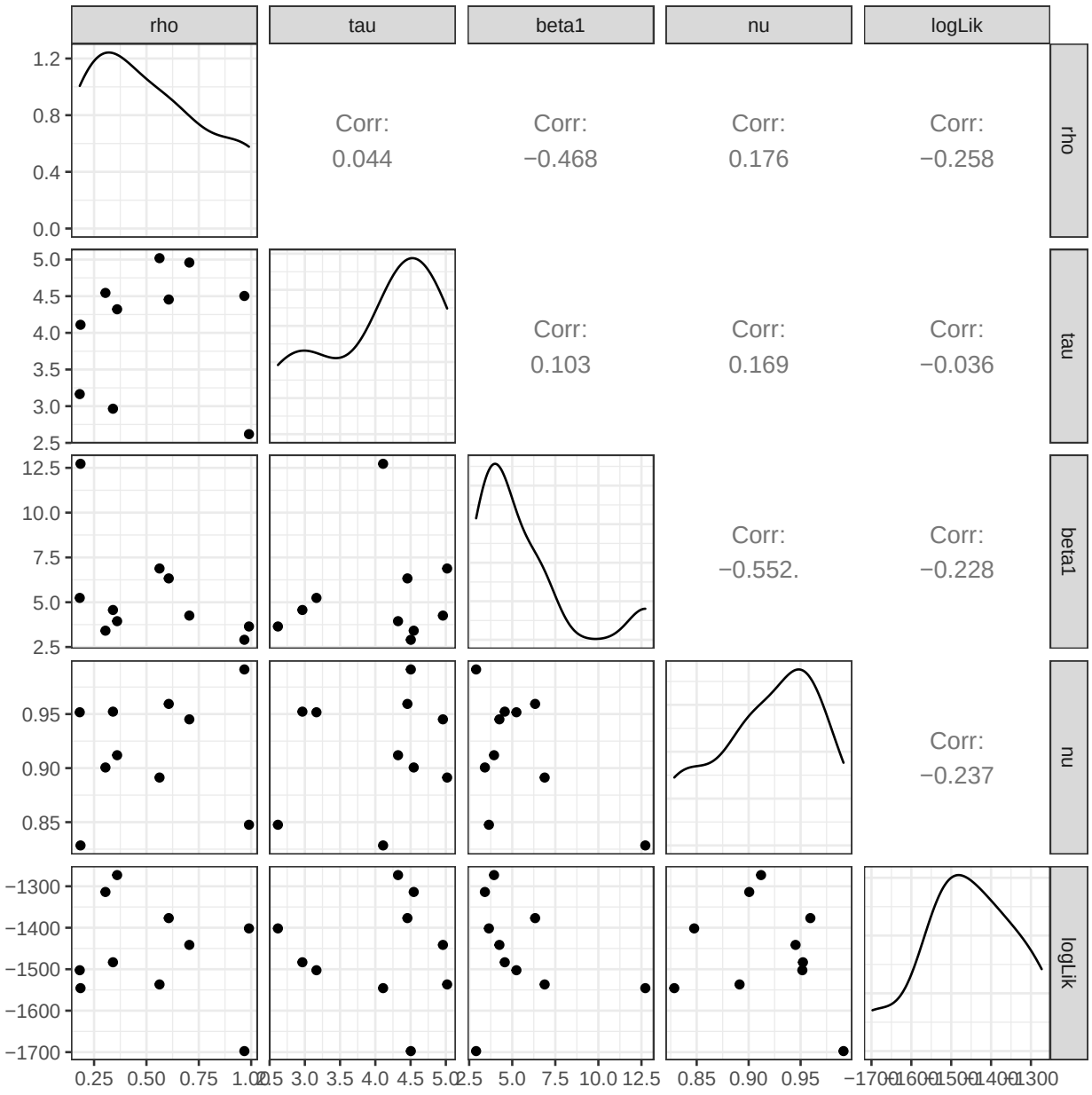

Figure S-2: Bivariate relationships between variables after fitting endemic period. Compare to S9 of Lee et al. (2020) supplement.

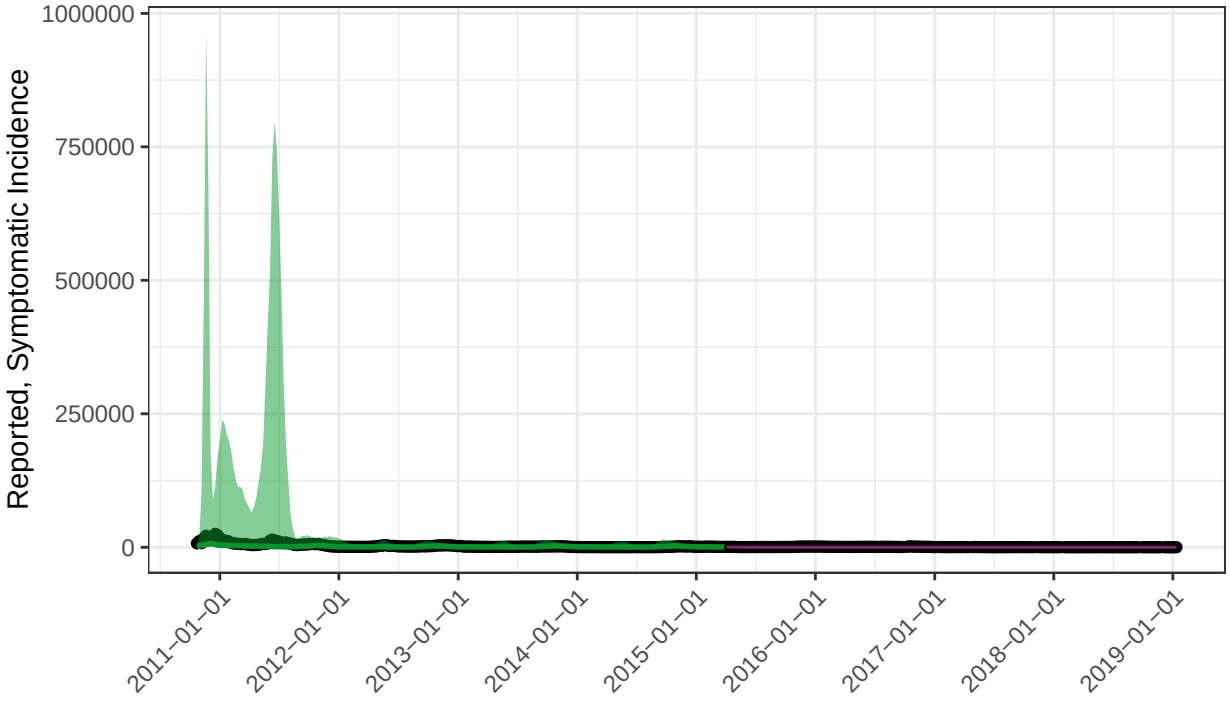

Figure S-3: Simulations from various parameter sets. Simulations from parameter sets with likelihood $<$ -2200 are removed. Compare to Figure S7 of Lee et al. (2020b).

simulations are shown in Figure S-3. While not a perfect replication of the results given in Lee et al. (2020b), the similarities between the model simulations, and the fact that we based our implementation of Model 1 primarily on the source code that was provided, makes us confident that we have been able to obtain a reliable representation of the model described in Lee et al. (2020a).

S7.2 Model 2 Replication

Model 2 was developed by a team that consisted of members from the Fred Hutchinson Cancer Research Center and the University of Florida (hereafter referred to as the Model 2 authors). While Model 2 is the only deterministic model we considered in our analysis, it contains perhaps the most complex descriptions of cholera in Haiti: Model 2 accounts for movement between spatial units; human-to-human and environment-to-human cholera infections; and transfer of water between spatial units based on elevations charts and river flows.

The source code that was used to implement Model 2 in Lee et al. (2020a) was written in the `Python` programming language, and is publicly available at `10.5281/zenodo.3360857` and its accompanying GitHub repository `https://github.com/lulelita/HaitiCholeraMultiModelingProject`. In order to perform our analysis in a unified framework, we re-implemented this model in the `R` programming language using the `spatPomp` package (Asfaw et al., 2021), which facilitates the creation of meta-population models. We note that the travel and water matrices used to implement the complex dynamics in Model 2 (Lee et al., 2020b) are not available in either the Zenodo archive

or the GitHub repository; instead, we obtained these matrices via personal correspondence with the Model 2 authors.

Using these matrices, and the point estimates for model parameters listed in (Lee et al., 2020b), we created trajectories of the cholera dynamics and compared this to available data. These trajectories, shown in Figure S-4, are very similar to the trajectories shown in Figure S15 of Lee et al. (2020b), suggesting that our re-implementation of Model 2 in the `haitipkg` is a faithful approximation to the original model used in Lee et al. (2020a).

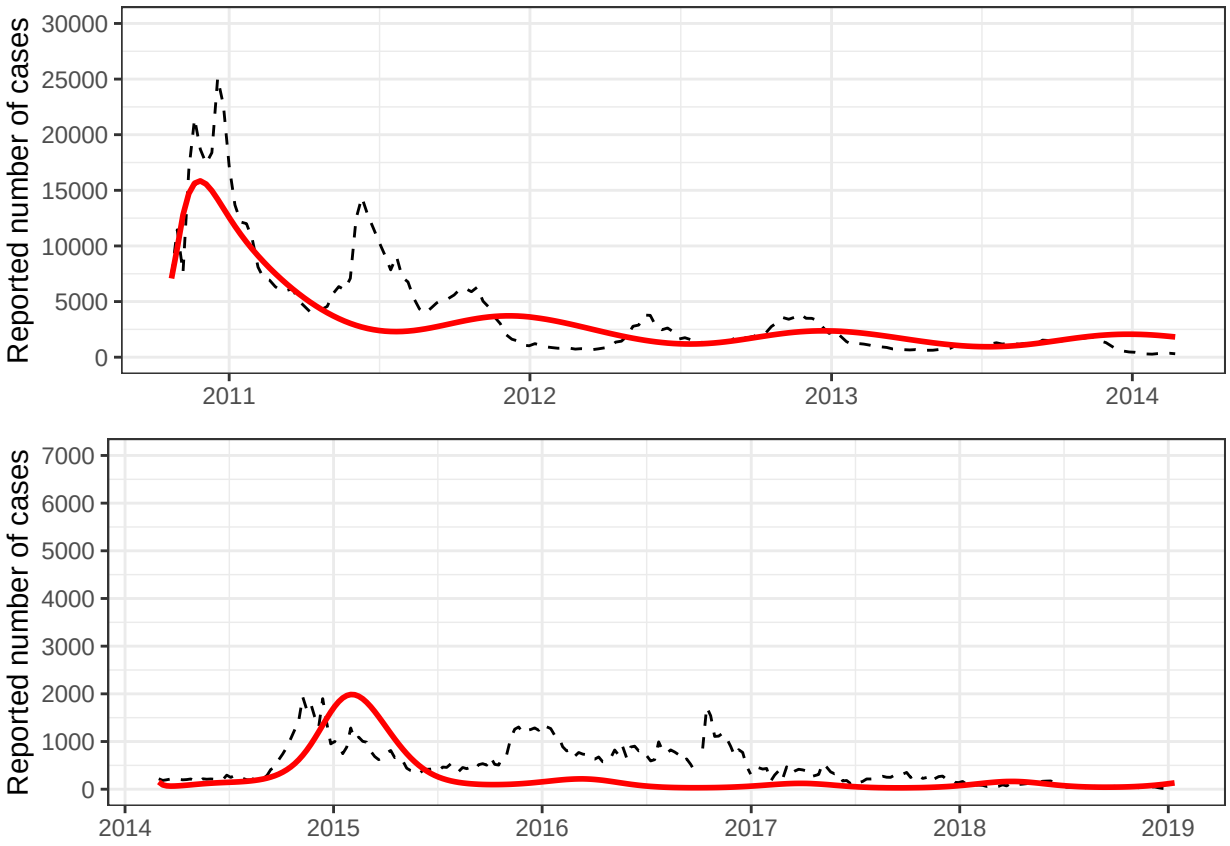

Figure S-4: Model 2 trajectories using the `haitipkg`. Compare to Figure S15 of Lee et al. (2020b).

We do note, however, several discrepancies exit between Figure S-4 and Figure S15 of Lee et al. (2020b), such as a delayed second wave in the epidemic period and a larger initial wave during the endemic period. While these are only minor discrepancies, because Model 2 is deterministic one may expect to be able to obtain perfect replication in model trajectories. We believe these discrepancies are a result of slight difference in model parameters. For example, we note that the parameters β, β_W (See Table S-1) had reported values of 9.9×10^{-7} and 4.03×10^{-2}, respectively (Table S13 of Lee et al. (2020b)), but were actually fit to data and therefore likely had values that were far more precise. Additionally, our implementation of Model 2 used a time scale of years and many of the parameters were reported on a weekly scale, so small differences may result due to unit conversions. The collective effect of these small differences in model parameters likely will result in differences in model trajectories, as seen here.

S7.3 Model 3 Replication

Model 3 was developed by a team of researchers at the Laboratory of the Swiss Federal Institute of Technology in Lausanne, hereafter referred to as the Model 3 authors. The code that was originally used to implement Model 3 is archived with the DOI: `10.5281/zenodo.3360723`, and also available in the public GitHub repository: `jcblemai/haiti-mass-ocv-campaign`. Because the code was made publicly available, and final model parameters were reported in the supplementary material of Lee et al. (2020a), we were able to reproduce Model 3 by directly using the source code. In Fig. S-5, we plot simulations from this model. This figure should be compared to Figure S18 of Lee et al. (2020a). We note that slight differences may be accounted for due to variance in the model simulations and the difference in programming language used to produce the figure.

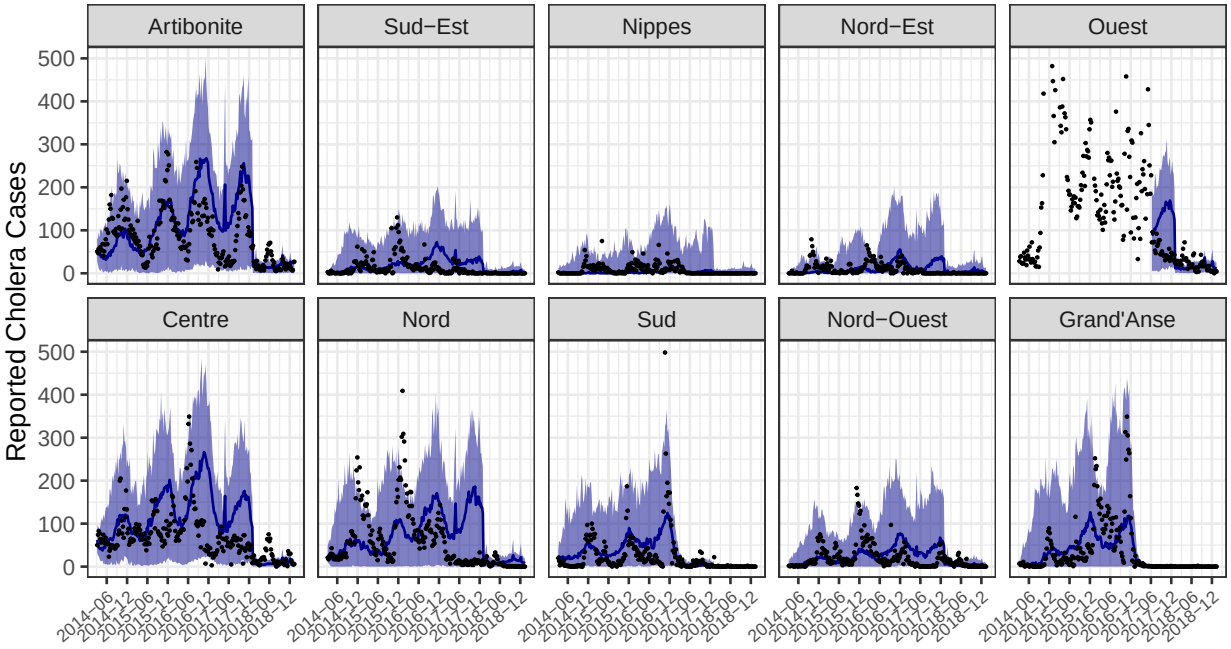

Figure S-5: Simulations from Model 3. Compare to Figure S18 of Lee et al. (2020a).

Supplementary References

Asfaw, K., Ionides, E. L., and King, A. A. (2021). `spatPomp`: R package for statistical inference for spatiotemporal partially observed Markov processes. `https://github.com/kidusasfaw/spatPomp`.

Bhadra, A., Ionides, E. L., Laneri, K., Pascual, M., Bouma, M., and Dhiman, R. C. (2011). Malaria in Northwest India: Data analysis via partially observed stochastic differential equation models driven by Lévy noise. *Journal of the American Statistical Association*, 106:440–451.

Bretó, C., He, D., Ionides, E. L., and King, A. A. (2009). Time series analysis via mechanistic models. *Annals of Applied Statistics*, 3:319–348.

Bretó, C. and Ionides, E. L. (2011). Compound Markov counting processes and their applications to modeling infinitesimally over-dispersed systems. *Stochastic Processes and their Applications*, 121:2571–2591.

Bretó, C., Ionides, E. L., and King, A. A. (2020). Panel data analysis via mechanistic models. *Journal of the American Statistical Association*, 115(531):1178–1188.

Domeyer, J. E., Lee, J. D., Toyoda, H., Mehler, B., and Reimer, B. (2022). Driver-pedestrian perceptual models demonstrate coupling: Implications for vehicle automation. *IEEE Transactions on Human-Machine Systems*, pages 1–10.

He, D., Ionides, E. L., and King, A. A. (2010). Plug-and-play inference for disease dynamics: Measles in large and small towns as a case study. *Journal of the Royal Society Interface*, 7:271–283.

King, A. A., Domenech de Cellès, M., Magpantay, F. M., and Rohani, P. (2015). Avoidable errors in the modelling of outbreaks of emerging pathogens, with special reference to ebola. *Proceedings of the Royal Society B: Biological Sciences*, 282(1806):20150347.

King, A. A., Nguyen, D., and Ionides, E. L. (2016). Statistical inference for partially observed Markov processes via the R package pomp. *Journal of Statistical Software*, 69:1–43.

Lee, E. C., Chao, D. L., Lemaitre, J. C., Matrajt, L., Pasetto, D., Perez-Saez, J., Finger, F., Rinaldo, A., Sugimoto, J. D., Halloran, M. E., Longini, I. M., Ternier, R., Vissieres, K., Azman, A. S., Lessler, J., and Ivers, L. C. (2020a). Achieving coordinated national immunity and cholera elimination in Haiti through vaccination: A modelling study. *The Lancet Global Health*, 8(8):e1081–e1089.

Lee, E. C., Chao, D. L., Lemaitre, J. C., Matrajt, L., Pasetto, D., Perez-Saez, J., Finger, F., Rinaldo, A., Sugimoto, J. D., Halloran, M. E., Longini, I. M., Ternier, R., Vissieres, K., Azman, A. S., Lessler, J., and Ivers, L. C. (2020b). Supplement to: Achieving coordinated national immunity and cholera elimination in Haiti through vaccination: A modelling study. *The Lancet Global Health*, 8(8):e1081–e1089.

Romero-Severson, E., Volz, E., Koopman, J., Leitner, T., and Ionides, E. (2015). Dynamic variation in sexual contact rates in a cohort of HIV-negative gay men. *American Journal of Epidemiology*, 182:255–262.

Stocks, T., Britton, T., and Höhle, M. (2020). Model selection and parameter estimation for dynamic epidemic models via iterated filtering: application to rotavirus in germany. *Biostatistics*, 21(3):400–416.

Subramanian, R., Romeo-Aznar, V., Ionides, E., Codeço, C. T., and Pascual, M. (2020). Predicting re-emergence times of dengue epidemics at low reproductive numbers: Denv1 in rio de janeiro, 1986–1990. *Journal of the Royal Society Interface*, 17(167):20200273.

Wale, N., Jones, M. J., Sim, D. G., Read, A. F., and King, A. A. (2019). The contribution of host cell-directed vs. parasite-directed immunity to the disease and dynamics of malaria infections. *Proceedings of the National Academy of Sciences*, 116(44):22386–22392.