Malaria Epidemiology, Dynamics and Control

A Framework for Robust Analytics and Adaptive Malaria Control

David L. Smith

2023-01-11

Contents

Foreword

A large fraction of my time over the past 20 years has been devoted to learning about malaria epidemiology, mosquito ecology, and vector control All the time, I was building and analyzing models of malaria epidemiology, dynamics and control. I've been looking for a way of organizing and applying the rich body of theory that has been developed. I wanted a framework that was *extensible* with *plug-and-play* modularity. I wanted the models to scale down to very fine-grained simulations, and to scale up to large-scale simulations. To serve the needs of malaria programs, we needed a way of discussing malaria as a changing baseline that was modified by control. To get exogenous forcing and vector control right, we needed to go all in on mosquito ecology. Sometime in the fall of 2022, the last few pieces came together, and it was time to write this book.

David L. Smith

Part I

Introduction

Chapter 1

Malaria Dynamics

Malaria is complex and heterogeneous, which makes it difficult to study and manage. A core limitation on making progress is the availability of information. Mathematical models are one way of dealing with all that complexity and making informed decisions despite the data gaps.

In basic research, we develop mechanistic models to understand malaria as a biological process. In malaria epidemiology, the states and parameters describe infection, immunity, transmission, disease, and drug taking in response to exposure. Scientists focus on basic biological mechanisms to understand differences in malaria across spectrum of transmission. Immunity and drug-taking are important factors to consider, but it is possible that parasite populations also differ across geographies, affecting malaria. A test of a model's validity is whether it can describe malaria accurately regardless of differences in drug taking patterns, the pattern of exposure, and other factors.

We study mosquito ecology and blood feeding to understand malaria transmission and develop theory for malaria control. Transmission models couple parasite infection dynamics in humans and mosquitoes through blood feeding. Mosquito populations are shaped by the aquatic habitats of immature mosquitoes – standing water bodies shaped by topography, hydrology, land use, and the chemistry of the rocks, soils, surrounding vegetation and polution. These habitats are filled (exogenously forced) by rainfall and after some eggs are laid, the mosquito dynamics are affected by crowding, predation, and other endogenous dynamics. Larval development and parasite development rates are modified by temperature. Adult mosquito activity rates are affected by temperature and relative humidity. Indoor residual spraying kills mosquitoes when they rest on a sprayed surface, usually after blood feeding or during the process of searching for a host. Insecticide treated nets protect humans from biting and kill some mosquitoes. By reducing the availability of potential blood hosts, nets can slow blood feeding in some contexts. Larval source management reduces immature population densities. By studying transmission, we can start to understand malaria as a changing baseline that has been modified by malaria control. We can study how malaria persists in populations over time, and how various factors can modify mosquito population dynamics and blood feeding and thereby suppress transmission. Transmission models help us to set intervention coverage targets based on thresholds.

In policy, we use these models with the expectation that – if we fit the models by adjusting parameters that affect how malaria works in some particular place – they *should* help us understand transmission in some particular context and make good decisions about what to do.

Frustratingly, the heterogeneity and the complexity conspire against us. We would like to be sure about how malaria works across settings before we start using the models to stratify populations, tailor interventions to context, or targeting the interventions. Instead, we must proceed without having satisfactory answers to some basic questions. The models make it possible to fill in the gaps in various ways and evaluate the consequences of having missing data, to assess what missing data would have the biggest effect on malaria, and (ideally) to find a way of filling the gap.

To understand malaria or to give policy advice, we must start simple and then add complexity, layer on layer. To deal with missing information, we start with generic models, and then add details to address concerns about some of the details that we hope to identify by studying the systems as we intervene. This approach – starting simple and then layering on complexity – makes it possible to learn as we go. A question is when it stops making sense to add realism to a model. A model that it too simple and abstract might help us understand the basic dynamics and give generic advice, but we would question the model's adequacy if it can not reproduce the patterns we care about in some particular place at some particular time. As a rule of thumb, a model should be just complex enough to *describe* the patterns we care about. Speaking practically, it's hard to know you've gone far enough unless, at some point, it's clear that you've gone a bit too far.

Over the past few years, we developed a new framework for building models that would make it possible to start simple and then build models of malaria transmission at any level of complexity. We wanted to be able to build in realism by adding complexity one feature at a time. Through this process we can create nested, hierarchical models in branching chains. At the ends of the chains, we might find highly realistic models that are, perhaps, overfit. (The cautions against overfitting play out differently if you can go out and collect new data.) We call the framework's ability to do this **scalability** and the resulting swarms have **scalable complexity**. To serve these goals, we wanted modular software with *plug-and-play* functionality and we needed a high degree of structural flexibility. Most of all, we wanted the framework to be extensible. The primary design phase is over, and the algorithms have been published in two software packages. This softare has dramatically lowered the costs of building and analyzing these complex, realistic models. We are currently extending the library of *base models*, which includes some simple or classical models that are instructive or of historical interest.

This book is being written to introduce the features of the framework (see Figure 1). The book lives in the RAMP-Model-Library, which was set up during the primary design phase. The RAMP-Model-Library is where we made all our design mistakes: was the software truly plug-and-play, and was the framework truly extensible? As the primary design phase came to a close, the library that was once the laboratory became a classroom and a museum. The library is being transformed into a resource for any developer who wants to add new base models to the library or add functionality. Most of all, it is being set up for the end user, someone in a malaria program or working with a malaria program who wants to use simulation based analytics to analyze policies. This book is structured into a set of lessons that teach concepts. Some of the concepts build on one another, and others take on new challenges. We combine these lessons into some examples where we show some algorithms to build models fit for purpose (Figure 1).

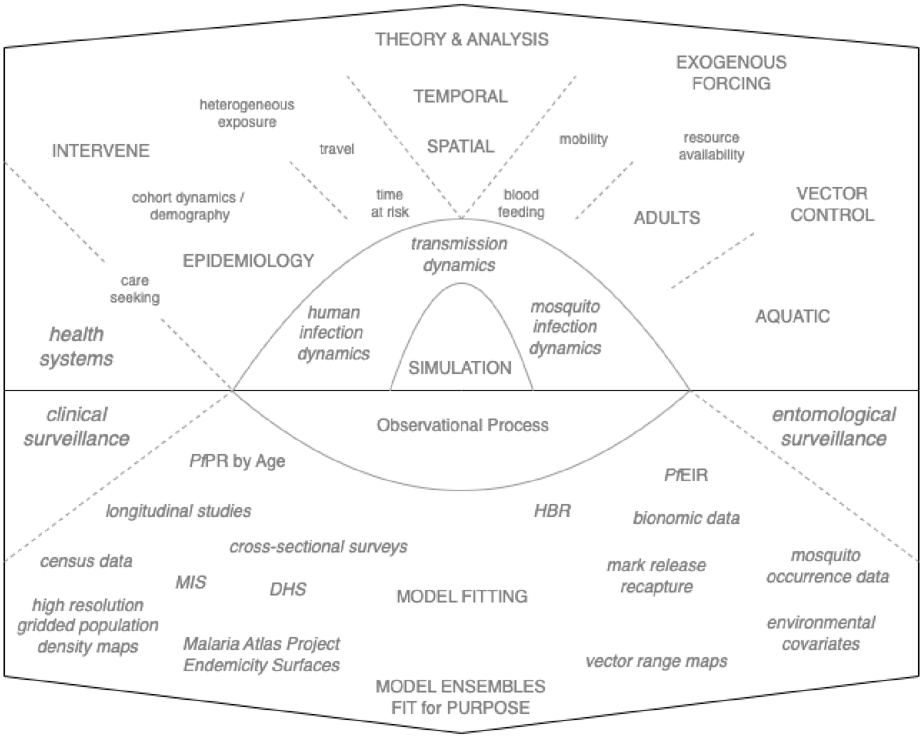


Figure 1.1: **Figure 1** A schematic diagram of the elements in the framework (top half) and the process of model building and model fitting (bottom half)

In malaria epidemiology (narrowly defined as a study of disease in humans), the relationship between exposure, infection, immunity, disease, and infectiousness changes in populations as they age, and it is affected by drug taking. This picture grows more complex as we consider vaccines, monoclonal antibodies, anemia, interactions with nutritional status, and human genetics. Our models need to interface with data from clinical settings and research, so they will need to consider diagnostics, parasite counts, detection, and transmission.

This is an overwhelming amount of complexity to graple with, so in this book, we start by studying some simple compartmental models. We will review classical queuing models for superinfection and the multiplicity of infection (MoI); new models for the age of infection (AoI) or stage of infection (SoI); immunity; parasite densities, fever, disease, and detection; gametocytes and transmission, and drug taking. To end up with models that can handle all the complexity, we build probabilistic models that combine these factors. In doing so, we find that we can do some powerful analysis, and we can map the states in these models onto outcomes that matter for research and policy: test positivity, parasite counts, infectiousness, and disease. With patience, we can combine these factors and develop a framework for understanding malaria in populations that match the features of individual-based simulation models. We end up with a sensible understanding malaria epidemiology as ontogeny and some very usable models that capture the the changing character of malaria in cohorts of humans as they age.

In this book, we start with a simple model for mosquito ecology and parasite infection dynamics in mosquitoes. We add aquatic population dynamics, mosquito population regulation, and exogenous forcing by weather. Later, we worry about adult mosquito behavioral states such as mating, sugar feeding, and egg laying. We introduce the concept of resource availability, and we develop an understanding of mosquito search and movement in response to resource availability. We take some deep dives to understand how mosquito spatial dynamics work at a fine spatial grain, and then we scale up to understand mosquito populations on landscapes.

At first, we describe mosquito blood feeding and transmission with a few simple parameters. Later, we develop a new model for mosquito blood feeding in a dynamically changing host population with parameters that allow host strata to be more or less available. We also modify our understanding of hetereogeneous exposure to biting. We develop a methods for modeling environmental heterogeneity, heterogeneous exposure by age, and a generalized way of handling failty – other sources of heterogeneous biting – through stratification.

We must take a detour to understand how to handle the effects of temperature on the parasite's extrinsic incubation period (EIP). We need a way of dealing with mosquito survival and dispersal through the EIP. This problem has been effectively solved.

To round out this picture, we need a way of dealing with other aspects of human ecology that affect malaria transmission dynamics, including human mobility, human demography, bed net usage, adherance to drugs, and care seeking. Differences among humans call for a synthesis of studies that have identified traits that affect malaria, stratification, and simulation to identify useful ways of propagating the heterogeneity through analyses.

To go along with a theory of transmission, we need a theory of control. We compute effect sizes and evaluate area effects. We develop a generalized concept of effect modification that considers the total effect of a single unit of control. We modify basic processes by including the effects of vector control and mass medical interventions (*e.g.* seasonal malaria chemoprotection, mass drug administration, vaccines, and monoclonal antibodies). Relying on behavioral state models and the concept of resource availability, we develop a models for integrated vector control.

In doing all this, we are building on an enormous body of work that started with Ronald Ross. While Ross is better known for identifying malaria parasites in a mosquito gut, proving that malaria is mosquito transmitted, we are more interested in the work he started later. After winning the Nobel Prize, Ross was instrumental in building solid quantitative foundations for malaria transmission and its measurement. Ronald Ross wrote the first models describing malaria transmission. In his writings from 1899 to 1908, it's clear that he's grappling for a set of equations that he can use to say something quite simple – if there are not enough mosquitoes, the malaria transmission can't be sustained. There must be a critical mosquito density, a cutoff that would define where malaria transmission can be sustained and where it can't. Ross was looking for a formula that encapsulated his intuition: how were thresholds related to the fact that it took two bites for a mosquito to complete its life cycle? Eventually, Ross wrote down some systems of equations that would describe malaria. The ideas, mathematics, and identification of parameters and processes were extended by other scientists later, most notably Alfred Lotka and George Macdonald.

It seems that the challenge of malaria control was what pushed Ross toward modeling. Ross's first model was a discussion of adult mosquito movement to guide larval source management [1]. The first model describing malaria transmission appeared in a book, *The Prevention of Malaria in Mauritius* [2]. When it came to thinking through control, Ross found it useful to do the math.

This is a book about how to do the math that is required for malaria programs. The goal is to use all the data available, but especially the data generated by malaria programs, to paint a clear picture of malaria transmission as a changing baseline that has been modified by control. The math is structured into three major domains: the humans and malaria epidemiology, including the effects of treating malaria with drugs; the mosquitoes and the way they have been changed by weather and vector control; and parasite transmission through mosquito blood feeding. Within each domain, there are multiple sub-domains, and there are built in ports and junctions to deal with heterogeneity and other features for malaria control. After a 140 years of studying malaria, there's a lot of detail that could be important in some way. Part of what we need to do is sort through all that detail to find what is most relevant.

The first model we present is a Ross-Macdonald model.

1.1 A Ross-Macdonald Model

The following is one version of the Ross-Macdonald model, which is the simplest way to get started [3]. This particular system of delay differential equations traces back to Joan Aron and Robert May [4].

We need to get through a lot of material, so background material in the following presentation is sparse. We assume the reader has the appropriate mathematical background. On occasion, there are links to vignettes that are designed to help fill in some of the gaps, if needed.

The simplest quantitive description of malaria dynamics tracks the number of infected and infectious mosquitoes and the number of infected and infectious humans. To develop systems of equations, we assign names to variables that represent these quantities: the number of infected people is denoted $X(t)$ (out of H total); the number of mosquitoes is $M(t)$; the number of infected mosquitoes is $Y(t)$ (out of $M(t)$ total); and the number of infectious mosquitoes is denoted $Z(t)$ (out of $M(t)$ total).

In dynamical systems, we ask how these variables change over time. For our first equation, we start with adult, female mosquito populations. (It is tiresome to repeat *adult, female* each time, and we're ignoring male mosquitoes at this point anyway, so *mosquito* hereafter means *adult, female mosquito*, unless we say otherwise.) The number of mosquitoes is changing as new adults emerge from aquatic habitats or die.

Mosquito Ecology

We assume the following:

- mosquitoes emerge at the rate of $\Lambda(t)$ adults, per day;
- mosquitoes die at a constant rate, g. The fraction surviving one day is e^{-g}; and the average lifespan is $1/g$.

Our first equation describes changes in the number of mosquitoes:

$$\frac{dM}{dt} = \Lambda(t) - gM \tag{1.1}$$

Infected Mosquitoes

Mosquitoes become infected after blood feeding on an infectious human. Here, we describe blood feeding using a few simple parameters.

To describe *blood feeding*, we assume the following:

- mosquitoes blood feed at the rate f, per mosquito, per day; in this model, this implies that the waiting time to a blood meal is $1/f$ days.
- a fraction of all mosquito blood meals, q, is taken on humans; we call this the *human fraction*
- the human blood feeding rate is the product of these two parameters, fq, which is defined as the number of human blood meals, per mosquito, per day.

The number of human blood meals by a population of vector mosquitoes, per person, per day is called the human biting rate (HBR). In this model, HBR is given by a formula:

$$\text{HBR} = \frac{fqM}{H}$$

Later, we can worry about how to map that onto an estimated HBR value, but not right now.

In the mosquito population, we need to know what fraction of blood meals end up infecting a mosquito that has not already been infected.

To describe *infection rates*, we assume the following:

- a fraction of human blood meals, infects mosquitoes. We call this quantity *net infectiousness* (NI) and (for reasons that we will discuss in a moment), we give it a name, κ:
$$\kappa(t) = c\frac{Xt}{H}$$
- infected mosquitoes die at the same rate as uninfected mosquitoes.

We can now write down our second equation describing changes in the number of infected mosquitoes:

$$\frac{dY}{dt} = fq\kappa(M - Y) - gY \tag{1.2}$$

Infectious Mosquitoes

To become infectious, a mosquito has to become infected and then survive through the extrinsic incubation period (EIP). We assume:

- mosquitoes become infectious after a fixed delay, τ days, called the EIP. The fraction of mosquitoes that survive through the EIP is $e^{-g\tau}$.
- infectious mosquitoes die at the same rate as other mosquitoes.

For a mosquito to become infectious, it must have become infected τ days ago and survived through τ days with probability $e^{-g\tau}$. To write this in equations, we use a subscripted τ to denote the value of a variable (M, Y or X) or term (κ) at time $t-\tau$. For example X_τ is the number of people who were infected and infecious at time $t-\tau$, and M_τ is the number of mosquitoes at time $t-\tau$.

The number of infectious mosquitoes that are added to the population at a point in time includes all the mosquitoes that became infected at time $t-\tau$ and survived the EIP. This is our third equation describing changes in the number of infectious mosquitoes:

$$\frac{dZ}{dt} = fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ \tag{1.3}$$

Here, Z represents the number of mosquitoes with *sporozoites* in their salivary glands. The *fraction* of mosquitoes with sporozoites in their salivary glands has been called the *sporozoite rate* (SR), which in our notation is

$$z = \frac{Z}{M}$$

The number of bites by vector mosquitoes, per person, per day is called the entomological inoculation rate (EIR). It is defined as the product of the HBR and the SR:

$$\text{EIR} = \text{SR} \times \text{HBR}$$

In our notation, the EIR is:

$$\text{EIR} = z\frac{fqM}{H} = \frac{fqZ}{H}$$

As with the HBR, we would like to know how to connect estiamted values of the EIR to our formulas. Since that's *really* complicated, we've spent a lot of time in the following sections discussing it.

Infected Humans

Humans become infected after being bitten by an infectious mosquito. We assume the following:

- A fraction b of all bites by infectious mosquitoes cause an infection.
- The hazard rate for infection, also called the *force of infection* (FoI) and denoted h is $b\times$ EIR:
$$h = fqb\frac{Z}{H}$$
- Infections clear at the rate r, per infection, per day (the average time to clear is $1/r$ days), and after clearing an infection a person becomes susceptible to infection again.

We can now write down our fourth equation describing changes in the number of infected humans:

$$\frac{dX}{dt} = h(H-X) - rX \tag{1.4}$$

Synopsis

While we presented these equations one at a time, they work as a system. To see it as a system, we write it here as four equations with two terms:

$$\begin{array}{ll} \frac{dM}{dt} & = \Lambda(t) - gM \\ \frac{dY}{dt} & = fq\kappa(M-Y) - gY \\ \frac{dZ}{dt} & = fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ \\ \frac{dX}{dt} & = h(H-X) - rX \\ \hline \kappa & = c\frac{X(t)}{H} \\ h & = bfq\frac{Z(t)}{H} \end{array} \tag{1.5}$$

The equations describe processes in three domains (Figure 2): adult mosquito population dynamics (M); parasite infection dynamics in mosquito populations (Y and Z); and parasite infection dynamics in human populations (X). The equations describing parasite infections in mosquito populations also include the variable M, so the mosquito infection dynamics are coupled to the mosquito population dynamics. The way we've written the equations, each compartment has an input term (*i.e.*, Λ, κ, or h) that depends on something else. We've passed Λ as a parameter. For the infection dynamics, the terms κ and h couple two separate systems. For adult mosquito dynamics, emergence is passed to the model as a parameters.

There are, of course, more compact ways of writing these equations. We have written the equations this way to emphasize a few things. First, the terms make it clear exactly how the equations in one domain are connected to another. Second, if we wanted to start *changing* some of the assumptions, these terms help to isolate the parts we might like to change. By writing the equations in this modularized form, we can start to understand how we might be able to write software that would allow us to represent mosquito infection dynamics with different systems of equations.

The next step is to find solutions.

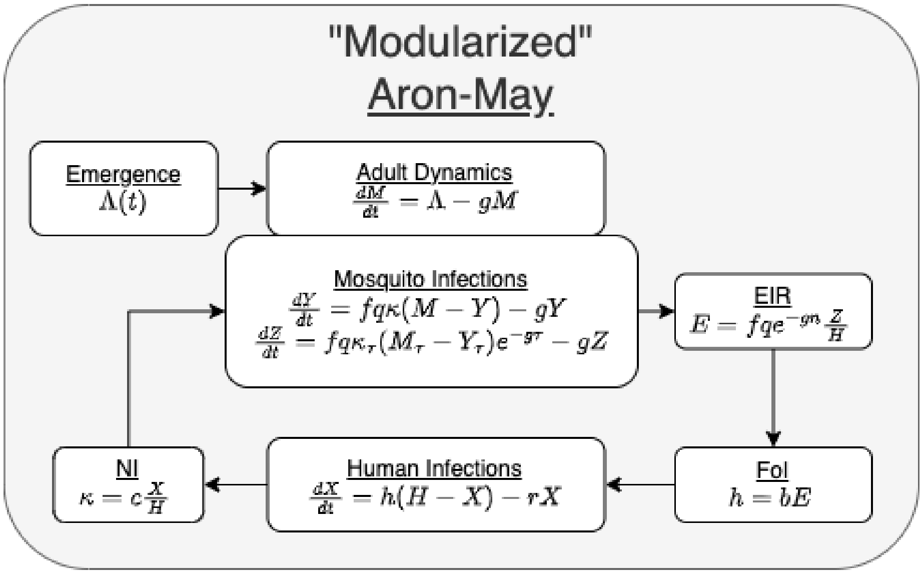

"Modularized"
Aron-May
Emergence
$\Lambda(t)$
Adult Dynamics
$\frac{dM}{dt} = \Lambda - gM$
Mosquito Infections
$\frac{dY}{dt} = fq\kappa(M-Y) - gY$
$\frac{dZ}{dt} = fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ$
EIR
$E = fqe^{-gn}\frac{Z}{H}$
NI
$\kappa = c\frac{X}{H}$
Human Infections
$\frac{dX}{dt} = h(H-X) - rX$
FoI
$h = bE$

Figure 1.2: **Figure 2** A diagram of the a version of the Ross-Macdonald model, using equations from Aron and May [4]

1.2 Solutions

What does a **solution** to these equations look like?

Solutions to these equations are values of the variables over time $(M(t), Y(t), Z(t), X(t))$ that satisfy the system of four equations described above. We call these solutions *orbits.* To put it another way, if we took the derivatives of the orbits for any variable at any point in time using the basic definition

$$\lim_{h\to 0} \frac{x(t+h)-x(t)}{h},$$

and then we used the values of the orbits at time t to compute dM/dt, dY/dt, dZ/dt, and dX/dt (*i.e.*, using the formulas on the previous page), they would be the same.

It is important that these orbits are unique: after specifying the *initial values* of these variables, there is one and only one solution. While we tend to produce orbits forward in time, the orbits are defined for all time – *i.e.* the process implies solutions far back into the past. These are deterministic equations, after all.

As written, the equations do not define a *model,* though it is an easy mistake to make (and one we'll probably make repeatedly when we can get away with it). Instead, the equations define a process or a **model family.** A model is something that *can* produce orbits, and we can't possibly produce orbits until we assign specific values to the parameters.

To find solutions of equations we use an R software package called `deSolve.` Because of the delay for the EIP, these are called *delay differential equations,* which are handled using a function called `dede`. An important part of these delay differential is that the values of variables at a time lag are retreived using a function called `lagvalue()`.

A very important part of solving a delay differential equation is that we must specify the initial conditions for an interval of time, not just at a point. (Since the equation for dZ/dt looks back τ units, we must specify values of $M(t)$, $Y(t)$, and $X(t)$ for all values of $t \in (-\tau, 0)$.) This forces an awkward choice, since we would need to know the solutions back in time to use them. What is typically done – and we've done it here – is to specify a constant set of initial values. Doing this introduces a little *numerical slop.* These values are *not* what we would get if we ran the equations backwards in time. We're happy to acknowledge this little problem and find ways around it, but we should never forget it is there.

1.2.1 Derivatives

With deSolve, solving differential equations is not difficult. The first step is to write down the equations to compute the derivatives.

```
dAronMay = function(t, y, params){with(params,{

  # Variables
  if(t<=tau) ylag<-y0 else ylag <- lagvalue(t-tau)
  M=y[1]; M_tau = ylag[1]
  Y=y[2]; Y_tau = ylag[2];
  Z=y[3];
  X=y[4]; X_tau = ylag[4]

  # Terms
  kappa = c*X/H; kappa_tau = c*X_tau/H
  h = b*f*q*Z/H

  # Dynamics
  dM = Lambda(t) - g*M
  dY = f*q*kappa*(M-Y) -g*Y
  dZ = f*q*kappa_tau*(M_tau-Y_tau)*exp(-g*tau) -g*Z
  dX = h*(H-X)-r*X

  return(list(c(dM, dY, dZ, dX)))
})}
```

1.2.2 Initial Values

To run the model, we must supply initial values. A useful convention for simple models is to pass the initial values as a named list. Later, we can turn the outputs into a data frame, and then we can retrieve the variables by name. If you're writing code yourself, remember that the initial values and the return value for the derivatives must occur in the same order.

```
y0= c(M=60, Y=0, Z=0, X=1)
```

1.2.3 Parameter Values

We pass the parameters as a list. It might seem like overkill, but we have written a function that takes default values and generates the list. This makes it easy to generate a new set of parameter values with alternative values, and it also helps us to write and pass function $\Lambda(t)$ with parameters we like. By passing the parameter as a list, the parameter values are available to the function `dAronMay` when we use `with(params, {})`.

Note that we have also attached the initial values of the variables as a parameter set, which are the return values for `lagvalue(t)` when `t<0`.

```
makeParams = function(y0,
                      g=1/12, f=1/2.5, q=0.95,
                      c=0.15,
                      b=0.55, r=1/200, H=1000,
                      m=.05, ss=1,
                      tau=10
                      ){
  ss = min(1,max(0, ss))
return(list(y0=y0,g=g,f=f,q=q,c=c,H=H,tau=tau,b=b,r=r,
  Lambda = function(t){m*H*(1 + ss*sin(2*pi*t/365))}))
}
params = makeParams(y0)
```

To make it absolutely clear, we are assuming:

- $g = 1/12$: mosquitoes live about 12 days, on average
- $f = 1/2.5$: mosquitoes feed every 2.5 days, on average
- $q = 0.95$: the human fraction is 95%; mosquitoes feed on humans 95% of the time
- $c = 0.15$: about 15% of bites on infectious humans infect a mosquito
- $b = 0.55$: about 55% of bites by infective mosquitoes cause an infection
- $r = 1/200$: human infections last about 200 days, on average
- $H = 1000$: we're simulating transmission in a population of a thousand humans
- $\tau = 10$: the extrinsic incubation period is about 10 days
- For emergence, we tune the average value using m and it is scaled to H:
  - The parameter m in the function above has been set to 0.05 by default.
  - The parameter ss affects the amplitude of the fluctuations. We force it to take on values between 0 and 1.
  - Emergence is modeled as a sinusoidal function with a yearly cycle.

$$\Lambda(t) = mH\left(1 + \sin\left(\frac{2\pi t}{365}\right)\right)$$

1.2.4 Solving

This code solves the equations:

```
require(deSolve)
tt = seq(0,5*365, by=5)
yout <- data.frame(dede(y=y0, times=tt, func=dAronMay, parms=params))
```

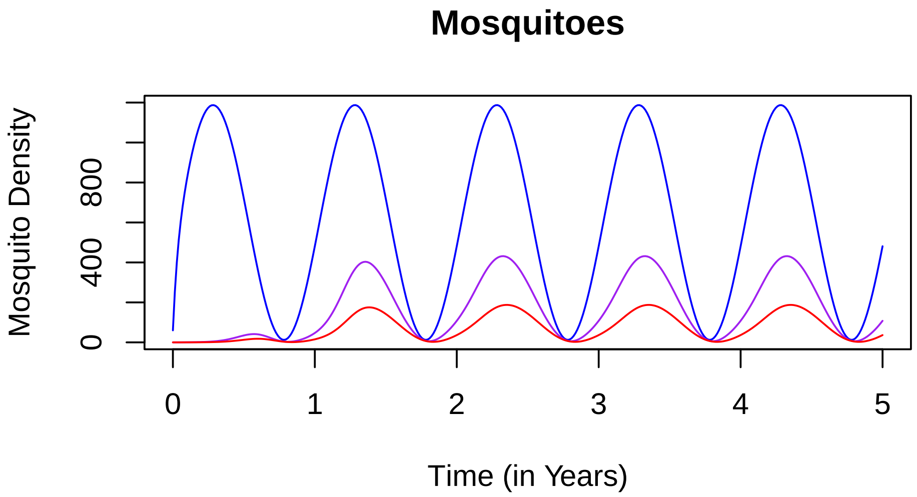
1.2.5 Visualizing

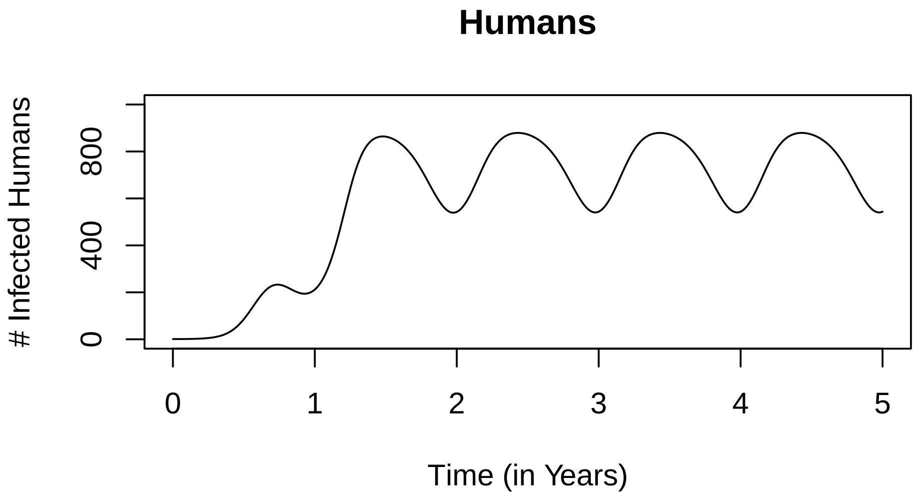
```
plotTS_AronMay = function(yout){with(yout,{
  par(mfrow = c(2,1))
  plot(time/365, M, type = "l", col = "blue",
       xlab = "Time (in Years)",
       ylab = "Mosquito Density",
       main = "Mosquitoes")
  lines(time/365, Y, col = "purple")
  lines(time/365, Z, col = "red")

  plot(time/365, X, ylim = c(0,1000), type = "l",
       xlab = "Time (in Years)",
       ylab = "# Infected Humans",
       main = "Humans")
})}
```

This code plots the outputs

```
plotTS_AronMay(yout)
```





1.3 The Basic Reproductive Number, R_0

1.4 exDE

We have written the software that solves these equations in a package called exDE.

Chapter 2

References

If you want a PDF and can't find it at the link provided, let us know and we can help you find a copy.

1. Ross R. The logical basis of the sanitary policy of mosquito reduction. Science. 1905;22: 689–699. doi:10.1126/science.22.570.689
2. Ross R. Report on the Prevention of Malaria in Mauritius. London: Waterlow; 1908. Available: https://play.google.com/store/books/details?id=0Mc1AQAAMAAJ
3. Smith DL, Battle KE, Hay SI, Barker CM, Scott TW, McKenzie FE. Ross, Macdonald, and a theory for the dynamics and control of mosquito-transmitted pathogens. PLoS Pathog. 2012;8: e1002588. doi:10.1371/journal.ppat.1002588
4. Aron JL, May RM. The population dynamics of malaria. In: Anderson RM, editor. The Population Dynamics of Infectious Diseases: Theory and Applications. Boston, MA: Springer US; 1982. pp. 139–179. doi:10.1007/978-1-4899-2901-3_5