

Applied Malaria Dynamics

Dynamical Systems for Adaptive Malaria Control

David L. Smith and the RAMP Team

2023-06-21

Contents

Foreword	5
Models for Policy	7
Contributors	13
 I Basic Concepts	 15
1 Malaria Dynamics	17
1.1 Aron and May's Equations	18
1.1.1 Mosquito Ecology	18
1.1.2 Blood Fed Mosquitoes	18
1.1.3 Infected Mosquitoes	19
1.1.4 Infectious Mosquitoes	20
1.1.5 Infected Humans	21
1.1.6 ... as a System	21
1.2 Solutions	22
1.2.1 Derivatives	24
1.2.2 Initial Values	25
1.2.3 Parameter Values	25
1.2.4 Solving	26
1.2.5 Visualizing	27
1.3 Steady States	28
1.3.1 Mosquito Density	28
1.3.2 EIR	28
1.3.3 Vectorial Capacity	30
1.3.4 Malaria Prevalence & Thresholds	31
1.3.5 Checking our Work	33
1.4 Stable Orbits	36
1.4.1 Thresholds	36
1.4.2 Orbits	37
1.4.3 Average Dynamics	38
1.5 Discussion	38
 2 Mosquito Dynamics	 39

2.1	Aquatic Dynamics	40
2.1.1	Equations	40
2.1.2	Solutions	40
2.1.3	Regulation	41
2.2	Understanding Mosquito Dynamics	43
II	Supplements	45
3	References	47

Foreword

A large fraction of my time over the past 20 years has been devoted to learning about mathematical models of malaria epidemiology, transmission dynamics, mosquito ecology, vector control, and the evolution of resistance. All the time I was building and analyzing models, I was looking for a way of organizing and applying the rich body of theory developed over more than a century of malaria research, starting with Ross and Macdonald [1]. A new framework would integrate the concepts and models that have influenced malaria through the present day [2, **ReinerRCJ2013SystematicReview?**], and it might even serve as a platform for recasting a theory of malaria epidemiology, transmission dynamics, and control [3].

The goal was to build a framework that could serve malaria policy. Policy discussions might need the math, but the discussion should never be about the math. There were some core challenges, but if we could solve those, we could make the math easier to use so that discussions would focus on the issues that mattered. Ideally, the framework could be taken up and used by teams of local experts working in their own countries to reduce the burden of malaria and plan for its elimination.

I wanted the new framework to be *extensible*, with *plug-and-play* modularity (for major dynamical components), and with structural flexibility. It should have the capability of scaling down for fine-grained spatial simulations [4–6], or scaling up to understand or analyze regional processes and the emerging patterns [7]. To serve the needs of malaria programs, a framework would need built-in support for exogenous forcing by weather and vector control to model malaria as a changing baseline modified by control. To get integrated vector control right, we went all in on mosquito ecology with an individual-based simulation model with exquisite biological detail [8], which inspired new ideas about mosquito search and a new base model for mosquito ecology and behavior. To serve programmatic needs, we needed algorithms that could address the durability of a unit of vector control – coverage could decay over time through loss (*e.g.* bed nets) or waning potency. To make all the pieces fit together, we needed interfaces that could connect up models in a generic way; in the design phase, we worked with two model families for each major dynamical component – one that was dead simple, and one that had a was highly realistic. In some cases, the interface designs called

for development of new algorithms: blood feeding, egg laying, environmental heterogeneity, human mobility, and mosquito dispersal. In making a master list to test the framework’s extensibility, we found that some odd cases that needed to pass information among components – endectocides and auto-disseminated larvicides – but it was easy enough to accommodate these. These models needed supporting theory. We wanted the software to help understand thresholds, so we wrote the routines that would compute thresholds for malaria transmission in heterogeneous systems, when appropriate.

Sometime in the fall of 2022, the last few pieces came together. We published the first versions of MicroMoB and exDE at CRAN. We submitted a paper to PLoS Computational Biology [9]. Then it was time to write this book.

With the software developed, we have the capability of building models that are up to the task of guiding malaria policy. The advantage of using the framework and accompanying software is that we found design solutions after discovering (usually the hard way) the potential pitfalls that arise when combining models. The framework took longer to develop than I had expected, in part, because there were more pitfalls than we had anticipated.

This book shows, through examples, how to use the software to build malaria models. Even without the software, it fills a gap for students who have taken an introduction to mathematical epidemiology or infectious disease modeling and want to go on in malaria. What are all the special topics that would need to be covered to build models that could be needed in malaria? This book *could* be the basis for such a course, if there were ever enough students. Since there will probably never be enough graduate students at my university who are interested in applied malaria dynamics, the material is being developed for any student anywhere. This book teaches how to build models for malaria policy, but it stops short of applying models to policy. That is covered in another book, **Robust Analytics for Malaria Policy**.

The premise of this book is thus that the reader has a solid background in infectious disease models and malaria. We assume they’ve seen the Ross-Macdonald model before, and that they know something about how to construct and analyze models. This book emphasizes concepts and teaches through examples. We have left out a lot of the technical and mathematical details, but we have written some vignettes and lessons to supplement the book. Most of this is found in the documentation for exDE or MicroMoB or it can be found in the **RAMP-Model-Library**.

I’ve done the primary writing for this book. The framework would not exist without the work of Sean L Wu and a few others. The book borrows from the work of others, and we have done our best to give credit through citations. It has been a collaborative process (see Contributors). The errors, however, are mostly mine. If you find mistakes or have questions, please drop me a note by email: smitdave@gmail.com.

– David L Smith

Models for Policy

Malaria is complex and heterogeneous, which makes it difficult to study and manage. A core challenge in both science and policy is the availability of information. Mathematical models can help us understand and analyze all that complexity and make informed decisions despite the data gaps.

There are good reasons why we might use different models in basic research and policy analytics. Basic research is epistemologically conservative, by design. In policy, decisions *must* be made in a timely way, and they *should* use all available evidence, even if it's weak. An advantage of policy is that, if we make the effort, we can identify key areas of uncertainty, identify priority data needs, and collect new data that could help resolve some of the most important sources of uncertainty. Building models to do this requires drawing heavily on basic research. In giving advice, we must give different weights to the uncertainty than we would in research.

In basic research, we develop mechanistic models to understand malaria as a biological process. In malaria epidemiology, the states and parameters describe infection, immunity, infectiousness, disease, and drug taking in response to exposure. Scientists focus on basic biological mechanisms in order to understand differences in malaria across spectrum of transmission. Immunity and drug-taking are important factors to consider, but it may be that differences in epidemiology and disease across settings arise from differences in the local parasite populations. The models are a way of summarizing knowledge in a quantitative form – something like a complex hypothesis. A test of a model's adequacy is whether it can describe malaria accurately after accounting for differences in drug taking patterns and pattern of exposure.

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 for immature mosquito populations. These habitats are standing water bodies, and they are shaped by topography, hydrology, land use, and the water chemistry, which is affected by surrounding rocks, soils, vegetation and pollution. 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, relative humidity, and vector control. Indoor residual spraying (IRS) 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 (ITNs) 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 (LSM) reduces immature population densities.

By studying mosquito ecology and malaria transmission dynamics, we can start to understand malaria as a changing baseline that has been modified by malaria control. This is the problem confronted daily in malaria programs, but dealing with the evidence requires having the tools available to synthesize data describing different parts of malaria. The models help translate evidence into information that can be used to make decisions, to make strategic plans, and to mark progress against national plans. The models encapsulate information about transmission in context, so it is possible to study how malaria persists in a place over time, and how various factors have modified (or could modify) mosquito population dynamics and blood feeding and thereby suppress transmission. Transmission models help us to set intervention coverage targets based on an understanding of malaria connectivity to surrounding regions and local 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 admit that we don't know everything we'd like to, and we probably never will. We must proceed with policy without having satisfactory answers to some basic questions. In policy, we will use the models to evaluate the consequences of having missing information, but we will also use the models to help us prioritize missing data so we can fill in the gaps. What missing data would reduce our uncertainty about what to do about malaria? How do we fill the critical knowledge gaps.

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 is 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 could not reproduce the patterns we cared 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 and *weigh* the relevant options to give advice. Practically speaking, 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 in policy given the urgency of acting in a timely way, but it is also possible to go out and collect new data.) We call the framework's ability to do this **scalability** and the resulting swarms have **scalable complexity**. The iterative attempt to make plans, weigh evidence, quantify uncertainty, gather new data to reduce uncertainty, and then restart the annual cycle, is called **adaptive malaria control**.

To make this possible, we needed a way of building models that would keep the focus on the policy questions and on a dialogue between malaria managers and the analytical support team. We thus sought to design modular software with *plug-and-play* functionality and a high degree of structural flexibility. We needed the framework to be extensible. After making a lot of mistakes, the primary design phase is over, and the algorithms have been published in two software packages. We are currently extending the library of *base models*, which includes some simple or classical models that are instructive or of historical interest. We are also fine-tuning the design requirements for models as we develop protocols that streamline fitting models to data. The software avoids the mistakes we made over the past few years, reuses models, and streamlines the model building process. We hope this software has dramatically lowered the costs of building and analyzing these complex, realistic models.

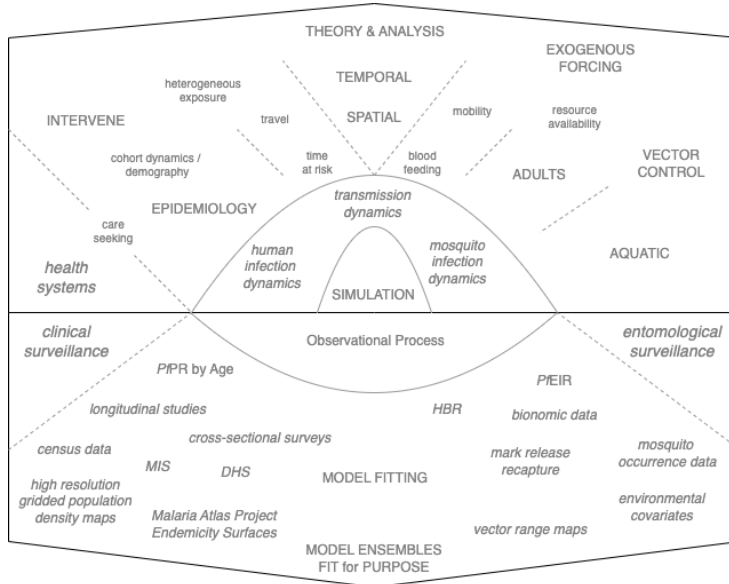


Figure 1: A schematic diagram of the elements in the framework (top half) and the process of model building and model fitting (bottom half)

This book has been written to introduce the features of the framework (see Figure 1.1). The book itself is embedded 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. When a topic deserves a deeper dive, we have supplemented this book with vignettes or lessons.

In malaria epidemiology (narrowly defined as a study of infection and 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 intervening with vaccines or monoclonal antibodies, or as we look at interactions with anemia, 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. Combining these factors can give rise to an overwhelming amount of complexity. Later, we will introduce new models and show how it possible to simplify all this complexity and make sense of malaria.

We are interested in using these models to guide policy, which requires both solid computation and good communication. In this book, we lay a foundation for understanding the complexity 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 – development of immunity as a part of an organisms history. We back this view with some very usable models that capture the changing character of malaria in cohorts of humans as they age.

We are interested in understanding malaria control in context, which requires delving into mosquito ecology and behavior. 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 heterogeneous exposure to biting. We develop a methods for modeling environmental heterogeneity, heterogeneous exposure by age, and a generalized way of handling faulty—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, adherence 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, which proved that malaria is mosquito transmitted, we are more interested in the academic work that followed.

After winning the Nobel Prize in 1902, 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 was searching for quantitative way of saying something simple – if there are not enough mosquitoes, the malaria

transmission can't be sustained. There must be a critical mosquito density, above the cutoff malaria transmission would be sustained, and below it malaria would be eliminated. 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 [10]. The first model describing malaria transmission appeared in a book, *The Prevention of Malaria in Mauritius* [11]. 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 software 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 design features 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.

We have organized the concepts in this book around a narrative that allows us to introduce the core concepts – those that make modular computation possible – in an order that minimizes the need to draw on unfamiliar concepts. We start with the Ross-Macdonald model, but our next task is to update the model for mosquito blood feeding.

Our philosophy has been to design a framework for model building that can be used by programs. The material in this book is designed to be used by non-experts too, so in this context, *model building* means applying a set of tools to computational tasks we wish our brains could do.

The software we have developed is meant to lower the costs of building and using models. We want programs to be focused on the decisions, the data, the concepts, and the analysis. As a metaphor, some students learn a numerical method for approximating $\sqrt{2}$ in school, but after learning it once, they stop worrying about *how* it is computed and they punch buttons into a calculator. Knowing how to compute something is sometimes useful, but worrying about how to compute it each time would interrupt the process that called for computing it. Instead, we punch the formula into a scientific calculator or any software

that does computation confident that the machine knows how to do it. In applying models, the same kind of logic applies. People need to understand the concepts, but like a calculator, the tools should hide the technical details that don't add to a discussion. The software we have developed is a reliable interface for calculations designed to support policy.

To learn how to use that software, we need to get through a lot of material. The background material in the following presentation is fairly sparse. We are trying to introduce just *enough* mathematics to teach users the critical concepts so they know what the software can do. We assume that the work will be done by teams that include a few people who understand the mathematics, who can guide others through the process. To fill in some of the gaps and technical, we have written (or can write) vignettes. On occasion, the text includes links to these vignettes for those who might find them useful. Please send suggestions about new vignettes to smitdave@gmail.com.

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

Contributors

This is a work in progress, so the list of contributors will change over time.

The software package `MicroMoB` was written by Sean L Wu, Sophie Liebkind, and David L Smith. The software package `exDE` was written by Sean L Wu and David L Smith.

Most of the content so far was written by David L Smith. Contributors from the RAMP Team include:

- ...please consult Dave if you would like a writing role.

David L. Smith

Part I

Basic Concepts

Chapter 1

Malaria Dynamics

We start by introducing a Ross-Macdonald model [1]. This particular model family is a system of delay differential equations that traces back to a 1982 book chapter written by Joan Aron and Robert May [12]. We think it's a good starting point.

We chose it because it is *extensible*. The variables represent population densities, which are used to compute proportions, like *prevalence*. The variables in many other versions of the Ross-Macdonald are proportions. In some models, we would like to be able to change the *total* number of hosts, but if the variables are proportions, these are the denominators. When a variables in equations described proportions, they are much more difficult to modify.

We chose it because it already includes mosquito ecology, so we can bypass a lengthy discussion of the limitations of the Ross-Macdonald model. While Macdonald's analysis and formulas are familiar, they were not complete [13,14]. We develop a formula for vectorial capacity that is consistent with the intent of the original, but our analysis of sensitivity to parameters includes effects on mosquito ecology [15]. (A lengthy and philosophical discussion of the history and its failings is planned.)

We chose it because it is *realistic*. Time does not appear in most versions of the Ross-Macdonald model: the equations are *autonomous*. These equations use time to drive a seasonal pattern: they are *non-autonomous*. Since we know we are interested in dealing with exogenous forcing, we start out with a model that is forced.

While the following model is basic, we recommend reading it, if only because we introduce concepts and conventions that are important for the software design.

1.1 Aron and May's Equations

The simplest quantitative 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 and infectious 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 the variables (*i.e.* M , Y , Z , and X) 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.

1.1.1 Mosquito Ecology

To model changes in M , we assume the following:

- mosquitoes emerge from aquatic habitats at the rate of $\Lambda(t)$ adults, per day;
- mosquitoes die at a constant rate, g . This is equivalent to assuming that the mosquito lifespan is exponentially distributed with a mean $1/g$. The fraction surviving one day is e^{-g} .

Our first equation describes changes in the number of mosquitoes:

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

1.1.2 Blood Fed Mosquitoes

At this point, we will take a detour and define a variable describing the density of mosquitoes that have blood fed at least once, V . After blood feeding, a mosquito is either gravid or *parous*, meaning its ovaries are distended from laying an egg batch. We do this, in part, because the fraction of mosquitoes that are parous is routinely collected, and because it gives us a chance to focus on blood feeding.

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, $f q$, 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:

$$\text{HBR} = \frac{f q M}{H}$$

Later, we discuss the correspondence between the HBR in models and data.

$$\frac{dV}{dt} = f q (M - V) - g V \quad (1.2)$$

We won't use V to describe the dynamics of infection, but we might find it useful to understand how parity changes in mosquito populations.

1.1.3 Infected Mosquitoes

Mosquitoes become infected after blood feeding on an infectious human. To model changes in Y , we extend the model of blood feeding to include infection. We need to know what fraction of blood meals end up infecting a mosquito that has not already been infected.

To model changes in Y , we need 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 :

$$\kappa(t) = c \frac{X(t)}{H} \quad (1.3)$$

- 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:

$$\frac{dY}{dt} = f q \kappa (M - Y) - g Y \quad (1.4)$$

1.1.4 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 infectious 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:

$$\frac{dZ}{dt} = fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ \quad (1.5)$$

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 estimated 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.

1.1.5 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 $b \times \text{EIR}$:

$$h = fq b \frac{Z}{H}$$

- 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:

$$\frac{dX}{dt} = h(H - X) - rX \quad (1.6)$$

1.1.6 ... as a System

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

$$\begin{aligned} \frac{dM}{dt} &= \Lambda(t) - gM \\ \frac{dY}{dt} &= fq\kappa(M - Y) - gY \\ \frac{dZ}{dt} &= fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ \\ \frac{dX}{dt} &= h(H - X) - rX \end{aligned} \quad (1.7)$$

$$\begin{aligned} \kappa &= c \frac{X(t)}{H} \\ h &= bfq \frac{Z(t)}{H} \end{aligned}$$

These equations describe processes in three domains (Figure 2.1):

- adult mosquito ecology (M , and perhaps V);
- parasite infection dynamics in mosquito populations (Y and Z);
- 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

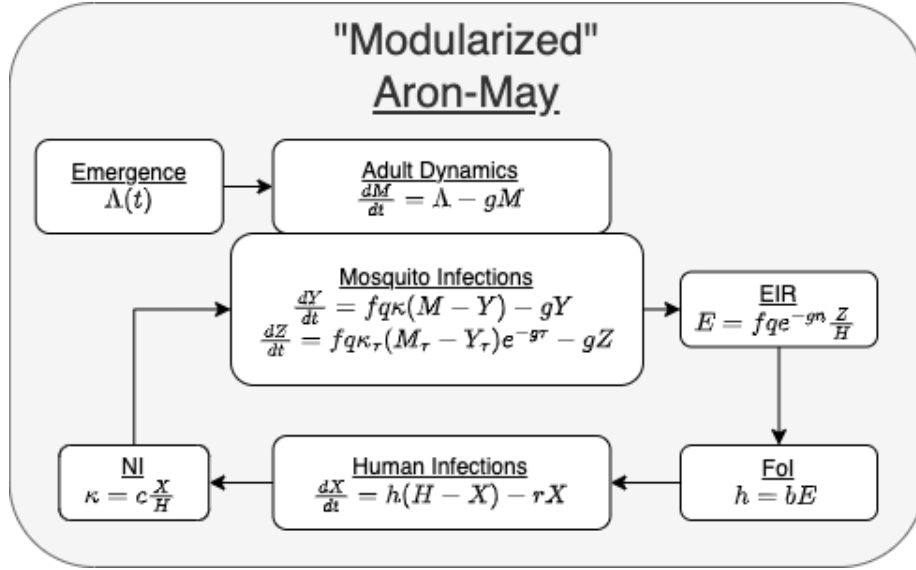


Figure 1.1: A diagram of the a version of the Ross-Macdonald model, using equations from Aron and May [12]

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.

NOTE: We don't introduce `exDE` or `MicroMoB` until [Modularity and Software].

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

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

and then we used the values of the variables at time t to compute dM/dt , dY/dt , dZ/dt , and dX/dt (*i.e.*, using the formulas), we would get the same values.

It is important that these orbits are unique: after specifying the *initial values* of the variables, there is one and only one set of orbits that solves the equations. When we solve the equations, we usually produce solutions from a starting point into a future, but the orbits are defined for all time – *i.e.* the process implies the existence of solutions far back into the past. These are deterministic equations, after all.

As written, the equations do not define a *model*. Instead, the equations define a process or a **model family**. A model is something that *can* produce orbits. A model is defined only after assigning specific values to the parameters. Informally, we will often slip and use the “model” to describe a model family. It’s easy to slip up, and sometimes we can get by with being sloppy, but we need to remember the distinction. When we say that the software is *modular*, we mean that it is easy to swap out one *model family* for another.

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 step in solving delay differential equations is a function `lagvalue()` that computes and returns the values of variables at a time lag, ℓ . In these equations, the lag is set by the EIP, τ , so we must evaluate `lagvalue(t-tau)`.

In solving *ordinary differential equations*, we must pass initial conditions. To solve a delay differential equations with a maximum lag ℓ , we must specify the initial conditions for the interval $[-\ell, t_0)$, where t_0 is the point in time when we start computing solutions. In these equations, 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 $t \in [-\tau, t_0)$. This forces an awkward choice, since we don’t know the solutions backwards in time, but would need to know those solutions to use them. What is typically done – and we’ve done it here – is to specify a constant set of initial values and moving on.

Doing this introduces a little *numerical slop*. By slop, we mean that these values are *not* what we would get if we ran the equations backwards in time. In these equations, it won’t affect our analysis most of the time, so we’re happy to acknowledge this little problem and find ways around it. It’s a little thing, but we should never forget it, because we might find that it *is* affecting our analysis at some point.

With `deSolve`, solving differential equations is not difficult – it just involves following a few steps. In the following, we walk through these steps:

- Write a function that computes the derivatives;
- Define initial conditions;
- Define the values of the parameters;
- Define a mesh on time;
- Call a function that solves the equations, such as `dede` for delay differential equations.

Many users will find that reading this code is like learning how to compute $\sqrt{2}$. If so, feel free to learn it once and then skip it.

1.2.1 Derivatives

The first step is to write down the equations to compute the derivatives. The solver expects a function with three required arguments (in this order):

- `t` is time
- `y` is the list of variables
- `params` is a set of parameters

The derivatives are computed and returned in the same order as ‘`y`’ in a `list`. To make code that is easy to read, we make `params` as a `list` with parameter names (see below), so that inside the function `with(params,{...})`, the parameter names are visible.

```
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. If you were writing code yourself, it would be important to remember that the initial values and the return value for the derivatives must occur in the same order.

A useful convention in $\{R\}$ 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.

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

The object `y0` is a named list – the names are attached but invisible.

```
y0
```

```
##  M  Y  Z  X
## 60  0  0  1
```

When we turn it into a list, with `as.list`, the names are attached to the values:

```
as.list(y0)$M
```

```
## [1] 60
```

If we use `with`, we create an environment where we can simply use the names:

```
with(as.list(y0), {
  M
})
```

```
## [1] 60
```

1.2.3 Parameter Values

We pass the parameters as a list. It might seem like overkill, but we have written a function `makeParams()` that takes default values and generates a 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))
  Lambda = function(t){m*H*(1 + ss*sin(2*pi*t/365))}
  return(list(y0=y0,g=g,f=f,q=q,c=c,
             H=H,m=m,tau=tau,b=b,r=r,Lambda=Lambda))
}
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

We define a mesh over time – the points in time when we would like to know the values of the variables:

```
tt = seq(0, 5*365, by=5)
```

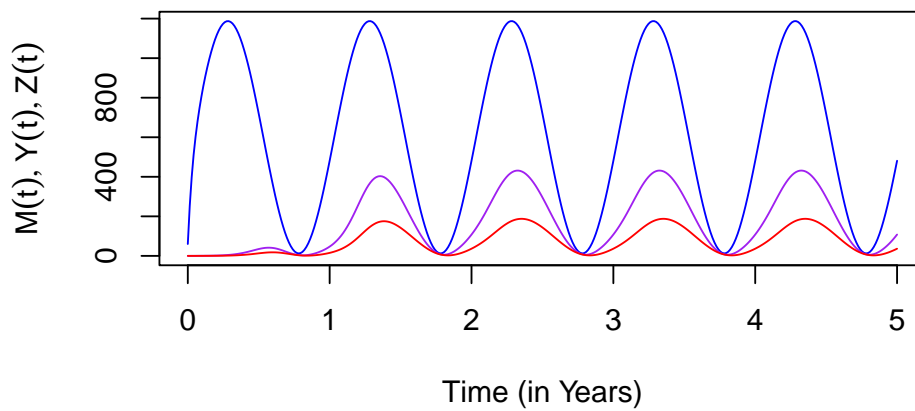
This code solves the equations:

```
require(deSolve)
yout <- dede(y=y0, times=tt, func=dAronMay, parms=params)
```

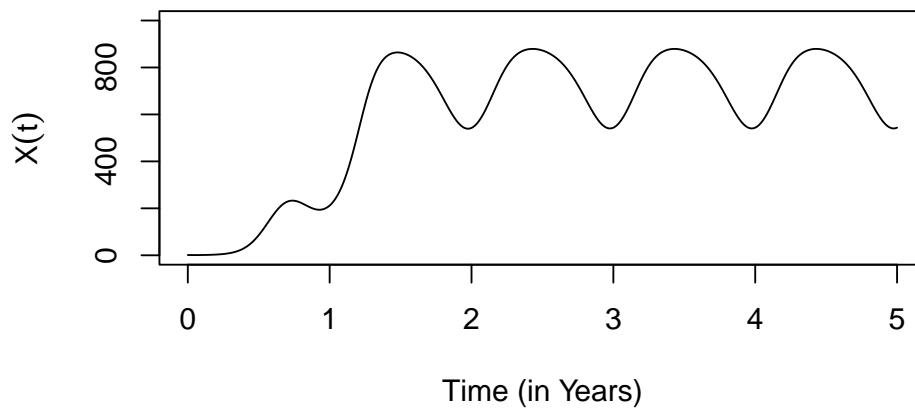
1.2.5 Visualizing

When we plot out the solutions, they look like this.

Mosquitoes



Humans



1.3 Steady States

Here, we analyze the system of equations in a narrow case when there is *no seasonality*, and the system reaches a *steady state*. To do so, we set the parameter $\mathbf{ss}=\mathbf{1}$, so that $\Lambda(t)$ is a constant; the resulting system is *autonomous*. We do this, in part, because the resulting system is easier to understand. We can develop intuition that can be applied (albeit with caution) to more complex systems. To be clear, we are dealing with this system:

$$\begin{aligned} \frac{dM}{dt} &= \Lambda - gM \\ \frac{dY}{dt} &= fq\kappa(M - Y) - gY \\ \frac{dZ}{dt} &= fq\kappa_\tau(M_\tau - Y_\tau)e^{-g\tau} - gZ \\ \frac{dX}{dt} &= h(H - X) - rX \end{aligned} \tag{1.8}$$

$$\begin{aligned} \kappa &= c \frac{X(t)}{H} \\ h &= bfq \frac{Z(t)}{H} \end{aligned}$$

As before, we have put the equations in their modularized form above, and the connecting terms below.

The first thing to note is that M affects Y and Z , which affect X ; but M is not affected by Y or Z . Mosquito population density is *exogenous* to malaria dynamics.

1.3.1 Mosquito Density

We can thus treat it separately in the analysis:

$$\frac{dM}{dt} = \Lambda - gM \tag{1.9}$$

Since emergence rates are steady, mosquito population density reaches a steady state when $dM/dt = 0$, which occurs at:

$$\bar{M} = \frac{\Lambda}{g} \tag{1.10}$$

1.3.2 EIR

Next, we note that at a steady state, the delayed values of variables and terms don't change, so from dY/dt , we get:

$$g\bar{Y} = fq\kappa(\bar{M} - \bar{Y}) \quad (1.11)$$

If we substitute the formula for \bar{M} and solve for \bar{Y} , we get:

$$\bar{Y} = \frac{fq\kappa}{g + fq\kappa} \frac{\Lambda}{g} \quad (1.12)$$

and since at the steady state, any variable at time $t + \tau$ is equal to its value at time t , we substitute the formula for $g\bar{Y}$ into dZ/dt to get:

$$g\bar{Y}e^{-g\tau} = g\bar{Z} \quad (1.13)$$

Solving for \bar{Z} we get:

$$\bar{Z} = \frac{fq\kappa}{g + fq\kappa} \frac{\Lambda}{g} e^{-g\tau} \quad (1.14)$$

At the steady state,

$$\text{EIR} = fq \frac{\bar{Z}}{H}.$$

In field studies, the EIR is the product of the HBR and the sporozoite rate (SR). The sporozoite rate (SR, z) is given by:

$$\bar{z} = \frac{Z}{M} = \frac{fq\kappa}{g + fq\kappa} e^{-g\tau} \quad (1.15)$$

So we can understand the EIR as having two parts:

$$\text{EIR} = \text{HBR} \times \text{SR} \quad (1.16)$$

or equivalently

$$\text{EIR} = \frac{fq\Lambda}{H} \times \frac{fq\kappa}{g + fq\kappa} e^{-g\tau} \quad (1.17)$$

This formula for the SR (albeit with slightly different notation) was originally derived as part of the Ross-Macdonald model [16, **ArmitageP1953NoteEpidemiology?**]. Also, Smith and McKenzie (2004) have written a useful discussion of mosquito demography [13].

1.3.3 Vectorial Capacity

Here, we pause to define a term that describes the number of human blood meals each mosquito would take over its whole life:

$$S = \frac{fq}{g}.$$

Since $1/g$ is the mosquito lifespan in days, and fq is the human blood feeding rate, S is the number of human bloodmeals a mosquito would take over its lifespan. Intuitively, it makes sense that this *should* be what drives transmission, since it takes two human blood meals to transmit malaria parasites.

If we rearrange the terms a bit, we can rewrite out the expression for the EIR:

$$\text{EIR}(\kappa) = fq \frac{\bar{Z}}{H} = \frac{\Lambda}{H} S^2 e^{-g\tau} \frac{\kappa}{1 + S\kappa} \quad (1.18)$$

This formula for the EIR has two parts. We call the first part *vectorial capacity* (V):

$$V = \frac{\Lambda}{H} S^2 e^{-g\tau} \quad (1.19)$$

The second part is an expression that involves mainly κ .

$$\frac{\kappa}{1 + S\kappa} \quad (1.20)$$

The relationship between VC and EIR at a steady state is a product:

$$\text{EIR}(\kappa) = V \frac{\kappa}{1 + S\kappa} \quad (1.21)$$

Vectorial capacity describes the slope of the EIR when κ is small:

$$d \frac{\text{EIR}(\kappa)}{d\kappa} \bigg|_{\kappa=0} = V \quad (1.22)$$

We say that VC describes *potential* transmission, even if the parasites are absent. Another way to say the same thing is that when κ is small, then:

$$\text{EIR}(\kappa) \approx V\kappa \quad (1.23)$$

We can interpret vectorial capacity (V) in simple terms. It describes *the number of infective bites that would arise from all the mosquitoes biting a single human*

on a single day but only if all those mosquitoes became infected. Vectorial capacity tells the story of potential parasite transmission by mosquitoes in four steps, which highlights the fact that two human blood meals are required for the parasite to be transmitted and complete its life-cycle.

$$\left| \begin{array}{c} \Lambda/H \\ \text{Mosquito} \\ \text{Emerges} \end{array} \right| \rightarrow \left| \begin{array}{c} S\kappa \\ \text{Parasite} \\ \text{Infects} \\ \text{Mosquito} \end{array} \right| \rightarrow \left| \begin{array}{c} e^{-g\tau} \\ \text{Mosquito} \\ \text{Survives} \\ \text{EIP} \end{array} \right| \rightarrow \left| \begin{array}{c} S \\ \text{Parasite} \\ \text{Infects} \\ \text{Human} \end{array} \right| \quad (1.24)$$

As a reminder, while Eq. (1.24) includes κ , the formula for VC, in Eq. (1.19), assumes that $\kappa = 1$: the VC describes transmission as if humans were perfectly infectious. It was defined this way on purpose: it was meant to include mosquito parameters and exclude human factors. We can think of VC as defining something like a conditional expectation, a maximum, or (as we have already said) a measure of *potential transmission by mosquitoes* that is independent of human factors.

While κ (the numerator in Eq.(1.20) accounts for *most* of the difference between the EIR and the VC, the rest of the difference is due to the denominator in Eq. (1.20), $1 + S\kappa$, which traces back to the formula from dY/dt , which assumes that mosquitoes are either infected or not. The denominator is a measure of saturation – the fraction of mosquitoes that get *superinfected* with parasites. The main point here is that as κ increases, saturation increases. If we set S to the values in the previous plots, we can isolate the relationship:

In these formulas, the measure of saturation is exactly $1 + S\kappa$. We could rewrite the relationship between the EIR and VC in a way that tells us something about how we might be underestimating a parasite's reproductive success:

$$\text{EIR} \times (1 + S\kappa) = V\kappa$$

which suggests that each infectious bite is passing along an excess $S\kappa$ bites.

1.3.4 Malaria Prevalence & Thresholds

We let x denote infection prevalence:

$$x = \frac{X}{H} \quad (1.25)$$

so $\kappa = cx$, and

$$\frac{dx}{dt} = \frac{1}{H} \frac{dX}{dt} = h(1 - x) - rx \quad (1.26)$$

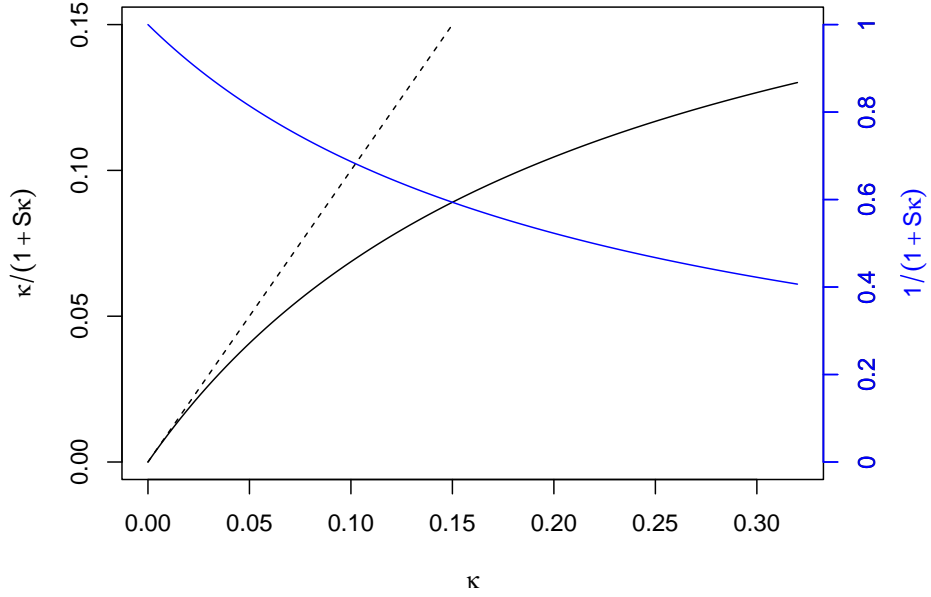


Figure 1.2: The effect (compare the solid and dashed black lines) and effect size of saturation (blue), graphically.

We can also define the basic reproductive number:

$$R_0 = \frac{bcV}{r}. \quad (1.27)$$

It is the product of four terms:

- Vectorial capacity, V , the number of infective bites, per person, per day;
- The number of days a person would remain infectious, $1/r$;
- The fraction of infectious bites that would infect a human, b ;
- The fraction of blood meals on infectious humans that would infect a mosquito, c

After taking their product, we can interpret R_0 as a measure of the parasite's reproductive success after a single generation. It only depends on where we start counting. It could be one of the following:

- the number of infected mosquitoes that would arise from a single infected mosquito;
- the number of infectious mosquitoes that would arise from a single infectious mosquito;

- the number of infected and infectious humans that would arise from a single infected and infectious human.

Here, R_0 plays an important role in these equations if we start with dX/dt ; then transform it to dx/dt ; then replace h with bE ; then replace κ with cx ; then divide by r ; and rearrange:

$$\frac{1}{r} \frac{dx}{dt} = x \left[R_0 \left(\frac{1-x}{1+cSx} \right) - 1 \right] \quad (1.28)$$

Since x is the prevalence, it is always in the interval $[0, 1]$. When x is very close to 0, then

$$\frac{1-x}{1+cSx} \lesssim 1. \quad (1.29)$$

and as x grows very small:

$$\lim_{x \rightarrow 0} \frac{1-x}{1+cSx} = 1. \quad (1.30)$$

It follows that when x is small, $dx/dt > 0$ if and only if $R_0 > 1$. Depending on R_0 , only one of two possibilities can hold:

- either $R_0 < 1$, so that $x = 0$ is the steady state;
- or $R_0 > 1$, and the steady state is:

$$\bar{x} = \frac{R_0 - 1}{R_0 + cS} \quad (1.31)$$

Since at the steady state, $\kappa = c\bar{x}$, we can plug this back into the formulas above to get \bar{Y} and \bar{Z} .

What we've learned about these equations is that if mosquito population densities are constant, then malaria reaches a steady state: if $R_0 > 1$, then there is a positive endemic equilibrium, and if $R_0 < 1$, then malaria is absent from the system. The system is said to be stable – in fact, is globally asymptotically stable, which means that all the orbits end up converging to the steady state. This statement has been proved many times in many papers, and since this book is focused on policy, we'll let others worry about proofs.

1.3.5 Checking our Work

An advantage of working in this environment is that we can check our work. One way we could solve these equations would be to run them for a very long time:

We make a parameter set that defines the model:

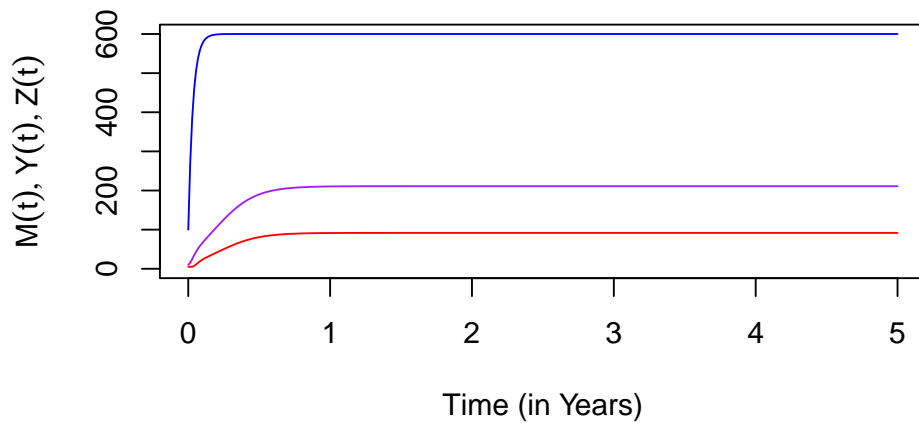
```
y0=c(M=100, Y=10, Z=5, X=200)
paramsSteady = makeParams(y0, ss=0)

dede(y0, times=tt, func=dAronMay, parms=paramsSteady) -> yout
tail(yout, 1)[-1] -> eq1
eq1
```

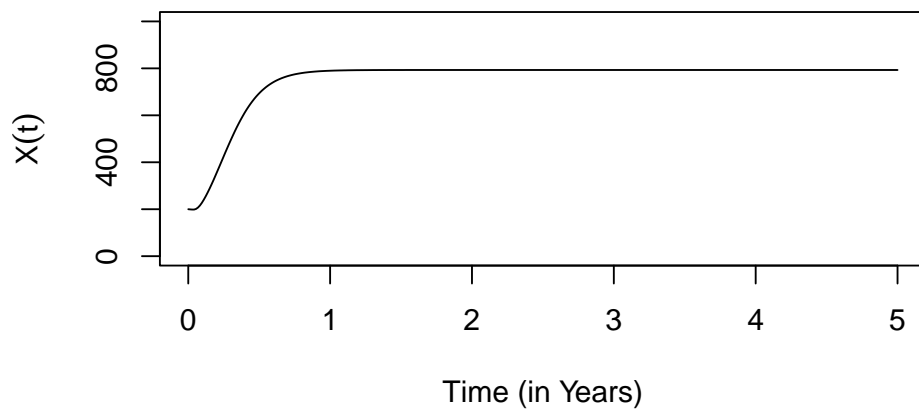
```
## [1] 600.00000 211.01706 91.70764 793.10541
```

By plotting it out, we can check to see if we've run it for long enough:

Mosquitoes



Humans



We can check our algebra by computing the same quantities, and R_0 and other quantities we care about:

```
steadyStates_AronMay = function(params){with(params,{
  Lambda = m*H
  Meq = Lambda/g
  S = with(paramsSteady, f*q/g)
  V = m*S^2*exp(-g*tau)
  R0 = b*c*V/r
  x = ifelse(R0>1,(R0-1)/(R0+c*S), 0)
  Xeq = x*H
  kappa = c*x
  Yeq = S*kappa/(1+S*kappa)*Meq
  Zeq = Yeq*exp(-g*tau)
  EIR = f*q*Zeq/H
  FoI = b*EIR
  aEIR = 365*EIR
  aFoI = 365*FoI
  extra=c(S=S, V=V, R0=R0, x=x, kappa=kappa,
          EIR=EIR, FoI=FoI)
  annual =c(aEIR = aEIR, aFoI=aFoI)
  list(std=c(M=Meq, Y=Yeq, Z=Zeq, X=Xeq),
       extra=signif(extra, 3),
       annual = signif(annual,3))
})}
steadyStates_AronMay(paramsSteady) -> eq2
eq2
```

```
## $std
##           M           Y           Z           X
## 600.00000 211.01706  91.70764 793.10541
##
## $extra
##      S      V      R0      x  kappa      EIR      FoI
## 4.5600 0.4520 7.4600 0.7930 0.1190 0.0348 0.0192
##
## $annual
## aEIR aFoI
## 12.7  7.0
```

Now, we can compare directly:

```
rbind(eq1=eq1, eq2=eq2$std)
```

```
##           M           Y           Z           X
## eq1 600 211.0171 91.70764 793.1054
## eq2 600 211.0171 91.70764 793.1054
```

1.4 Stable Orbits

If emergence rates vary seasonally, how much of the analysis that we did to understand *steady states* still holds? Obviously, if conditions are changing seasonally, the model does not reach a steady state. In fact, after modification to suit the context, many of the same principles translate. The steady state analysis provides a good qualitative guide, but that the answers will look different. Here, we illustrate by solving systems to illustrate some basic points, which is easy enough. Analysis of the resulting dynamics can be quite difficult; it is covered in [Temporal Dynamics].

1.4.1 Thresholds

There is a threshold condition $R_0 > 1$ that determines whether malaria is endemic, but the formula for R_0 depends on the form of $\Lambda(t)$. If we set $R_0 = 1$, we can show that the threshold for persistence in a seasonal environment is $R_0 > \sigma > 1$ (see Figure 1.1). The math to compute threshold conditions in seasonal environments is in [Temporal Dynamics].

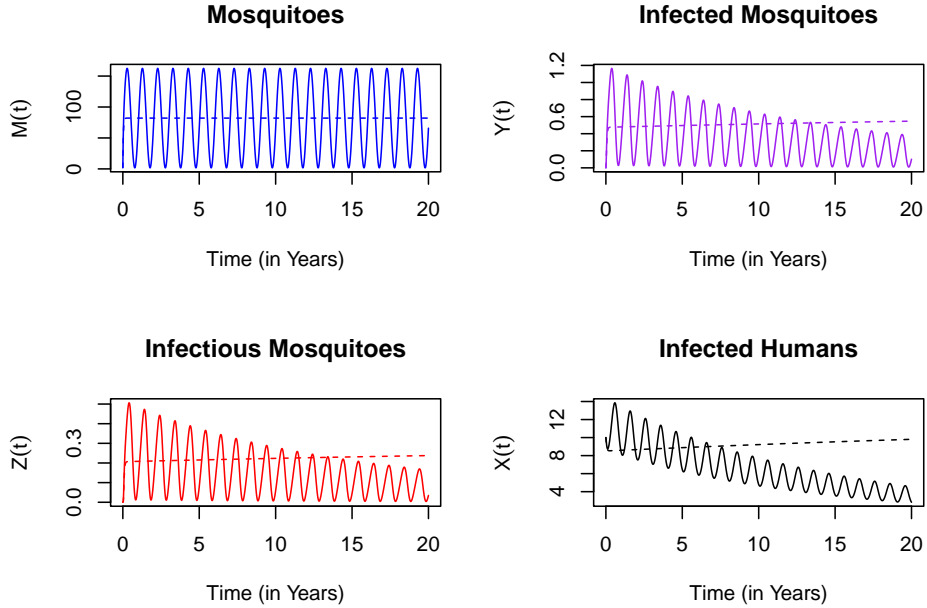


Figure 1.3: Here, we set $R_0 = 1.02$ for the model with constant emergence, and we show that malaria persists. For the same parameters and for the same *average* emergence rate, malaria declines with seasonality.

1.4.2 Orbits

If $R_0 > 1$, then all orbits converge to a set of *stable orbits* (See Figure 1.1). If $\Lambda(t)$ has an annual cycle, then after the orbits converge:

- $M(t+365) = M(t)$;
- $Y(t+365) = Y(t)$ and $Z(t+365) = Z(t)$;
- $X(t+365) = X(t)$.

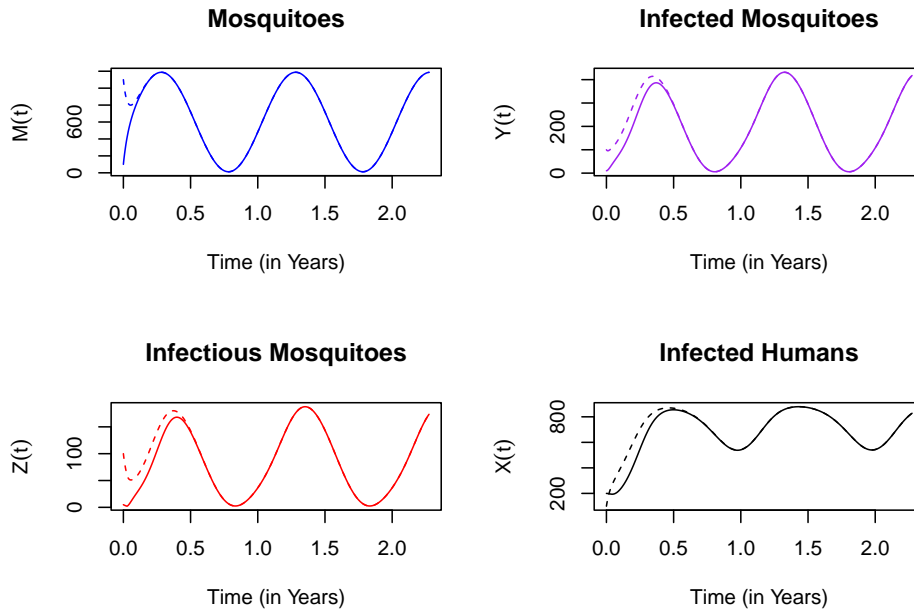


Figure 1.4: With different initial values, the orbits converge and eventually lie on top of one another.

1.4.3 Average Dynamics

If $R_0 > 1$ and malaria is endemic, the *average* prevalence of malaria infection is variable in a seasonal environment. While the prevalence is higher at the peak, the average for the whole year tends to be lower.

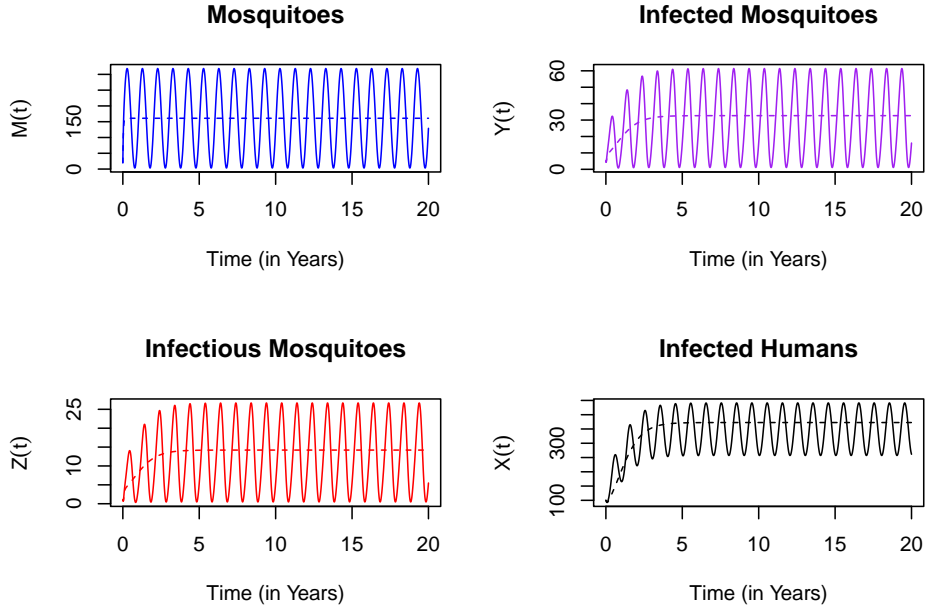


Figure 1.5: Here, we set $R_0 = 2$ for the model with constant emergence, and we show that the prevalence of malaria is similar in the seasonal environment, but it's higher as transmission peaks, lower in the off-season, and lower overall.

1.5 Discussion

If we were simply learning the math, ...

When we analyze these equations to determine their *stability* and to identify *threshold conditions*, we focus on threshold conditions and the behavior of these systems when malaria is rare. In most places, malaria is endemic so we need to be concerned about malaria immunity and its effects on transmission; malaria is under some level of control; and because of weather and other factors, the baseline conditions change from year to year.

Chapter 2

Mosquito Dynamics

In the previous chapter, we consider adult mosquito population density as a process closely related to, but also exogenous to the process of parasite transmission. We formulated the models for mosquito density in terms of an emergence rate, $\Lambda(t)$. In many studies, this might be good enough. There are some challenges in vector control, however, that call for a deeper understanding of adult mosquito population dynamics in relation to the population dynamics of immature mosquito populations. It's hard to imagine giving any advice about LSM without a model of mosquito ecology. Adult vector control will reduce mosquito population densities, but it would also reduce egg laying affecting immature mosquito populations [15]. Mosquito populations have their own thresholds for persistence, and their own spatial dynamics.

While it could be argued that these are not the primary concerns for malaria programs – we would not disagree – our mathematical framework must be strong enough to support analytics for integrated vector control, including LSM. Immature mosquito dynamics are one of the core dynamical components in our framework. We introduce the topic here, using a very basic model for mosquito ecology [17].

To extend the previous model, we need to make several changes and additional assumptions:

- We define terms that describe egg-laying by adult mosquitoes;
- We write a basic equation to describe the dynamics of immature mosquitoes in aquatic habitats as a function of egg-laying. These models describe how eggs hatch, how mosquito larvae develop and mature in aquatic habitats, pupate, and then emerge as adults. The models do not need to be complex, but they could be.
- We replace the parameter $\Lambda(t)$ from the Aron-May model with a term that describes emergence of adults from aquatic habitats.

One of the big themes we want to introduce here is that the outcomes of vector control can vary substantially by context because of differences in mosquito ecology.

2.1 Aquatic Dynamics

2.1.1 Equations

Here, we start with the equation for adult ecology from before, leaving out parasite transmission dynamics. We modify the basic slightly by adding one extra term that describes delayed maturation, a different response to crowding [17]. The mosquito maturation rate, $\psi e^{-\sigma L}$ defines the average time from egg to emergence, and we add one parameter that slows down larval development in response to crowding. The per-capita mortality rate, $\psi + \theta L$, has two parts: a density independent term ψ and a response to mean crowding θL . We assume that adults lay χ eggs per feeding cycle.

$$\begin{aligned} \frac{dM}{dt} &= \Lambda - gM \\ \frac{dL}{dt} &= \eta - (\psi e^{-\sigma L} + \phi + \theta L)L \\ \Lambda &= \frac{1}{2} \psi L e^{-\sigma L} \\ \eta &= \chi f M \end{aligned} \tag{2.1}$$

2.1.2 Solutions

```
BasicAquatic_dML = function(t, y, params){with(c(params, as.list(y)),{

  # Terms
  Lambda = psi*exp(-sigma*L)*L/2
  eta = chi*f*M

  # Dynamics
  dM = Lambda - g*M
  dL = eta - (psi*exp(-sigma*L) + phi + theta*L)*L

  return(list(c(dM, dL)))
})}

makePar_BasicAquatic = function(f=1/2.5, g=1/10, chi = 50, psi = 1/10, phi = 1/10, theta=1, sigma=1){
  list(f=f, g=g, chi=chi, psi=psi, phi=phi, theta=theta, sigma=sigma)
}
```


2.1.3 Regulation

```
tt = seq(0,100, by=1)
y0 = c(M=1, L=1)
params = makePar_BasicAquatic()
params1 = makePar_BasicAquatic(phi=1)
params2 = makePar_BasicAquatic(theta=1/10)
params3 = makePar_BasicAquatic(sigma=1/100, theta=0)
```

This code solves the equations:

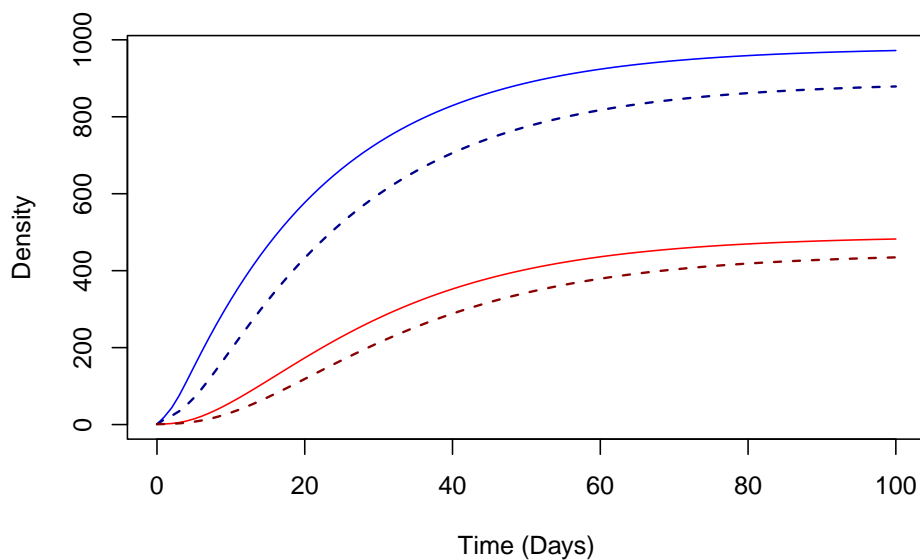
```
require(deSolve)
yout <- lsode(y=y0, times=tt, func=BasicAquatic_dML, parms=params)
yout1 <- lsode(y=y0, times=tt, func=BasicAquatic_dML, parms=params1)
yout2 <- lsode(y=y0, times=tt, func=BasicAquatic_dML, parms=params2)
yout3 <- lsode(y=y0, times=tt, func=BasicAquatic_dML, parms=params3)

plotML = function(out, llwd=1, clrL = "blue", clrM="red"){with(data.frame(out),{
  plot(time, L, type = "l", lwd=llwd, col = clrL, xlab = "Time (Days)", ylab = "Density")
  lines(time, M, col = clrM, lwd=llwd)
}})

addML = function(out, llty = 2, llwd=1, clrL = "blue", clrM="red"){with(data.frame(out),{
  lines(time, L, col = clrL, lty=llty, lwd=llwd)
  lines(time, M, col = clrM, lty=llty, lwd=llwd)
}})
```

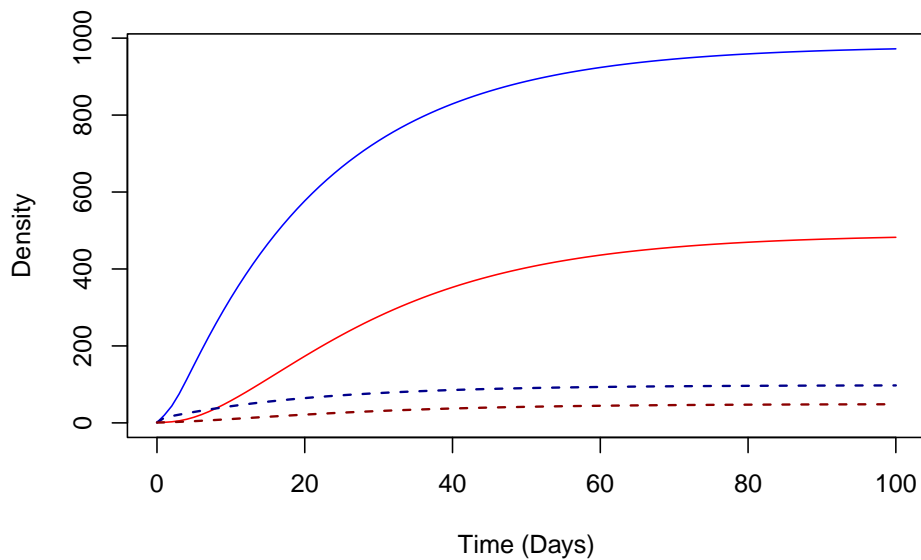
The first thing to point out is that changing the dynamics take some time to approach equilibrium – a couple of months, in this model. The second is that increasing mortality rates for mosquitoes in aquatic habitats by a factor of 10 affects the mosquito densities in minor ways.

```
plotML(yout)
addML(yout1, clrL = "darkblue", clrM="darkred", llwd=1.5)
```



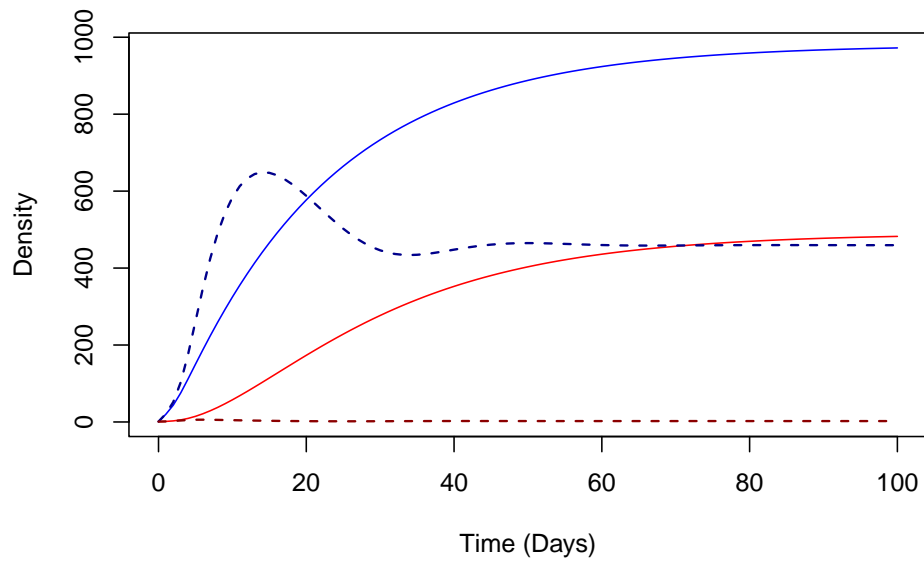
The third is that increasing the response to mean crowding has a very strong effect on the population dynamics.

```
plotML(yout)
addML(yout2, clrL = "darkblue", clrM="darkred", llwd=1.5)
```

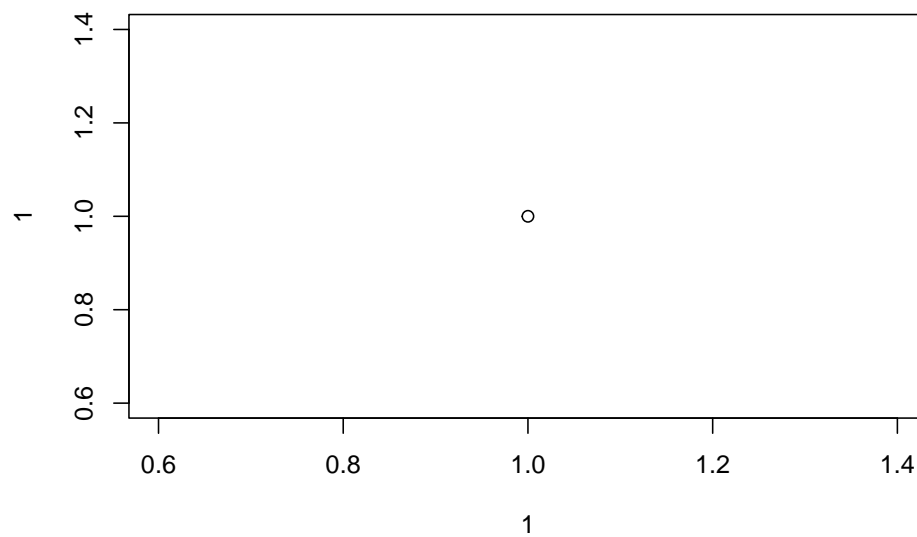


The third point is that density dependent responses to mean crowding cause much larger changes to this system than density-dependent mortality. Aquatic dynamics have a strong tendency to oscillate, but more critically, delayed maturation strongly affects the ratio of adult mosquitoes to adults.

```
plotML(yout)
addML(yout3, clrL = "darkblue", clrM="darkred", llwd=1.5)
```



```
plot(1,1)
```



2.2 Understanding Mosquito Dynamics

This

Part II

Supplements

Chapter 3

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. 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
2. Smith NR, Trauer JM, Gambhir M, Richards JS, Maude RJ, Keith JM, et al. Agent-based models of malaria transmission: A systematic review. Malar J. 2018;17: 299. doi:10.1186/s12936-018-2442-y
3. Smith DL, Perkins TA, Reiner RC Jr, Barker CM, Niu T, Chaves LF, et al. Recasting the theory of mosquito-borne pathogen transmission dynamics and control. Trans R Soc Trop Med Hyg. 2014;108: 185–197. doi:10.1093/trstmh/tru026
4. Carter R. Spatial simulation of malaria transmission and its control by malaria transmission blocking vaccination. International Journal for Parasitology. 2002;32: 1617–1624. doi:10.1016/S0020-7519(02)00190-X
5. Gu W, Killeen GF, Mbogo CM, Regens JL, Githure JJ, Beier JC. An individual-based model of *Plasmodium falciparum* malaria transmission on the coast of Kenya. Trans R Soc Trop Med Hyg. 2003;97: 43–50. doi:10.1016/s0035-9203(03)90018-6
6. Perkins TA, Scott TW, Le Menach A, Smith DL. Heterogeneity, mixing, and the spatial scales of mosquito-borne pathogen transmission. PLoS Comput Biol. 2013;9: e1003327.
7. Tatem AJ, Smith DL. International population movements and regional *Plasmodium falciparum* malaria elimination strategies. Proc Natl Acad Sci U S A. 2010;107: 12222–12227. doi:10.1073/pnas.1002971107

8. Wu SL, Sánchez C HM, Henry JM, Citron DT, Zhang Q, Compton K, et al. Vector bionomics and vectorial capacity as emergent properties of mosquito behaviors and ecology. *PLoS Comput Biol.* 2020;16: e1007446. doi:10.1371/journal.pcbi.1007446
9. Wu SL, Henry JM, Citron DT, Mbabazi Ssebuliba D, Nakakawa Nsumba J, Sánchez C. HM, et al. Spatial Dynamics of Malaria Transmission. 2022. Available: <http://medrxiv.org/content/early/2022/11/15/2022.11.07.22282044.abstract>
10. Ross R. The logical basis of the sanitary policy of mosquito reduction. *Science.* 1905;22: 689–699. doi:10.1126/science.22.570.689
11. Ross R. Report on the Prevention of Malaria in Mauritius. London: Waterlow; 1908. Available: <https://play.google.com/store/books/details?id=0Mc1AQAAMAAJ>
12. 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
13. Smith DL, McKenzie FE. Statics and dynamics of malaria infection in *Anopheles* mosquitoes. *Malaria Journal.* 2004;3: 13. doi:10.1186/1475-2875-3-13
14. Smith DL, Musiime AK, Maxwell K, Lindsay SW, Kiware S. A New Test of a Theory about Old Mosquitoes. *Trends Parasitol.* 2021;37: 185–194. doi:10.1016/j.pt.2020.10.011
15. Brady OJ, Godfray HCJ, Tatem AJ, Gething PW, Cohen JM, McKenzie FE, et al. Adult vector control, mosquito ecology and malaria transmission. *International Health.* 2015;7: 121–129. doi:10.1093/inthealth/ihv010
16. Macdonald G. The analysis of the sporozoite rate. *Trop Dis Bull.* 1952;49: 569–586. Available: <https://www.ncbi.nlm.nih.gov/pubmed/14958825>
17. Smith DL, Perkins TA, Tusting LS, Scott TW, Lindsay SW. Mosquito Population Regulation and Larval Source Management in Heterogeneous Environments. *PLOS ONE.* 2013;8: e71247. doi:10.1371/journal.pone.0071247