

1 Equations

The equations solved by `Mosquito23` are based on the following

$$\frac{\partial X(a, s, g, m)}{\partial t} = -d_a X + b_{s,g,m} \left[\overbrace{X(a-1, s, g, m)}^{a>0} - \overbrace{X}^{a<N-1} \right] + \overbrace{\nabla \cdot (D_{s,g,m} \nabla X - \mathbf{V}_{s,g,m} X)}^{a=N-1} +$$

$$a = 0 \quad \left\{ \begin{array}{l} \left(\frac{K - \sum_{m^*} \alpha_{m,m^*} \sum_{s,g} \sum_{i < \max(N,1)} X(i, s, g, m^*)}{K} \right) \times \\ \sum_{\substack{g_M, g_F, \\ m_M, m_F}} h(m_M, m_F, m) \lambda(g_F, m_F) i(g_M, g_F, g) \frac{w(m_M, m_F) X(N-1, M, g_M, m_M)}{\sum_{g^*, m^*} w(m^*, m_F) X(N-1, M, g^*, m^*)} \times \\ p(g_M, g_F, m_M, m_F, s) X(N-1, F, g_F, m_F) \end{array} \right.$$

where a = age (0 = newborn, $N-1$ = adult, $[1, N-2]$ = intermediate stages),

s = sex (M or F),

g = genotype (w wildtype, Gw heterozygous GM or GG homozygous GM),

m = mosquito species ($C = An. coluzzii$, $G = An. gambiae$)

and $X(a, s, g, m)$ is abbreviated to X .

2 Parameters

Parameter name	Explanation	Degrees of freedom																
d_a	Death rate $\begin{cases} d_{\text{adult}} & \text{for } a = N - 1 \\ d_{\text{larvae}} & \text{for } a < N - 1 \end{cases}$	2																
b	Larval emergence rate has distribution $\Gamma(k = N, \theta = b/N)$	1																
D	Diffusion rate	1																
\mathbf{V}	Advection vector field	Distribution of flight times																
K	Carrying capacity for species (spatially explicit, dependent on rainfall)	1 (for now)																
$h(m_M, m_F, m)$	Hybridisation rate (following Beeton et al. 2019) $= \begin{cases} 1 & \text{if } m = m_M = m_F \\ 0 & \text{otherwise} \end{cases}$	0 (may be set in code)																
λ	Larvae per female	1																
$i(g_M, g_F, g)$	Inheritance of genotype where $g = \{ww, Gw, GG\}$ <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td>ww</td> <td>Gw</td> <td>GG</td> </tr> <tr> <td>ww</td> <td>$\{1, 0, 0\}$</td> <td>$\{\frac{1}{2}, \frac{1}{2}, 0\}$</td> <td>$\{0, 1, 0\}$</td> </tr> <tr> <td>$Gw$</td> <td>$\{\frac{1}{2}, \frac{1}{2}, 0\}$</td> <td>$\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$</td> <td>$\{0, \frac{1}{2}, \frac{1}{2}\}$</td> </tr> <tr> <td>$GG$</td> <td>$\{0, 1, 0\}$</td> <td>$\{0, \frac{1}{2}, \frac{1}{2}\}$</td> <td>$\{0, 0, 1\}$</td> </tr> </table>		ww	Gw	GG	ww	$\{1, 0, 0\}$	$\{\frac{1}{2}, \frac{1}{2}, 0\}$	$\{0, 1, 0\}$	Gw	$\{\frac{1}{2}, \frac{1}{2}, 0\}$	$\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$	$\{0, \frac{1}{2}, \frac{1}{2}\}$	GG	$\{0, 1, 0\}$	$\{0, \frac{1}{2}, \frac{1}{2}\}$	$\{0, 0, 1\}$	0
	ww	Gw	GG															
ww	$\{1, 0, 0\}$	$\{\frac{1}{2}, \frac{1}{2}, 0\}$	$\{0, 1, 0\}$															
Gw	$\{\frac{1}{2}, \frac{1}{2}, 0\}$	$\{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$	$\{0, \frac{1}{2}, \frac{1}{2}\}$															
GG	$\{0, 1, 0\}$	$\{0, \frac{1}{2}, \frac{1}{2}\}$	$\{0, 0, 1\}$															
$p(g_M, g_F, s)$	Proportion of offspring of given sex given genotypes of parents $= \begin{cases} p(Gw, g_F, F) & = \frac{1}{2} (\frac{1}{a} - 1) \\ p(GG, g_F, F) & = \frac{1}{2} (\frac{1}{a} - 1) \\ 0.5 & \text{elsewhere} \end{cases}$	1 (accuracy a)																
$w(m_M, m_F)$	Relative probability of female of species m_F mating with male of species m_M $= \begin{cases} w(C, C) = w(G, G) = 1 \\ w(C, G) = w(G, C) = w \end{cases}$	1 (w) (defaults to $w = 0$)																
$\alpha(m, m')$	Lotka-Volterra competition between species	Defaults to $\alpha = I$ (may be set in code)																

3 Mosquito23

`Mosquito23` simplifies the above equations, by assuming that the diffusion and advection is handled by other parts of the code, so that the above equations reduce to a system of ODEs. To interface with `Mosquito23`, you need to know the ordering of its vector X . For species m , genotype g , sex s and age a , this is

$$X(m + gN_{\text{species}} + sN_{\text{species}}N_{\text{genotypes}} + aN_{\text{species}}N_{\text{genotypes}}N_{\text{sexes}}), \quad (1)$$

where

- N_{species} is the number of species,
- $N_{\text{genotypes}} = 3$ is the number of genotypes (with ordering $ww = 0$, $Gw = 1$, and $GG = 1$)
- $N_{\text{sexes}} = 2$ is the number of sexes (with order male = 0 and female = 1)
- N_{ages} (denoted by N in the above sections) is the number of ages (with ordering newborn = 0 and adult = $N_{\text{ages}} - 1$, with intermediate stages in between these).

The next sections write the equations explicitly and add some explanation.

3.1 Adults

All adult populations are governed by

$$\frac{dX(N-1, s, g, m)}{dt} = -d_{\text{adult}}X(N-1, s, g, m) + bX(N-2, s, g, m) . \quad (2)$$

Here $N-1$ indicates the adult age bracket, and the $N-2$ is the eldest juvenile age bracket. The first term on the right-hand side describes the mortality of adults, while the second term describes aging from the eldest juveniles. If $N=1$ there are no adults, and all populations are governed by the “Newborn larvae” equations, below.

3.2 Intermediate juveniles

For $0 < a < N-1$, the populations are governed by

$$\frac{dX(a, s, g, m)}{dt} = -d_{\text{larvae}}X(a, s, g, m) + b(X(a-1, s, g, m) - X(a, s, g, m)) . \quad (3)$$

For $N \leq 2$ there are no such intermediate juveniles. The first term on the right-hand side describes the mortality of this age-bracket of juveniles, while the term involving b describes aging to/from older/younger age brackets

3.3 Newborn larvae

For $a=0$, the populations are governed by

$$\frac{dX(0, s, g, m)}{dt} = -d_{\text{larvae}}X(0, s, g, m) - bX(a, s, g, m) + B(s, g, m) . \quad (4)$$

The first term describes mortality of newborns, while the second describes aging into the next age-bracket of juveniles. The final term describes the birth of newborn larvae. It is

$$B(s, g, m) = L\left(1 - \frac{C(m)}{K}\right) \sum_{g_M, g_F, m_M, m_F} P_{\text{offspring}}(s, g, m | g_M, g_F, m_M, m_F) P_{\text{mating}}(g_M, m_M, m_F) \lambda X(N-1, F, g_F, m_F) \quad (5)$$

This equation deserves explanations.

- $C(m)$ is the competition that a newborn feels from the rest of the larval populations. It is

$$C(m) = \sum_{a=0}^{N-2} \sum_{s=0}^{N_{\text{sexes}}-1} \sum_{g=0}^{N_{\text{genotypes}}-1} \sum_{m'=0}^{N_{\text{species}}-1} \alpha_{m, m'} X(a, s, g, m') . \quad (6)$$

Notice that this does not include adults $a = N-1$. If $N=1$, it is assumed that the carrying-capacity still applies, and the sum over a runs from 0 to 0. The Lotka-Volterra matrix α accounts for inter-specific competition. It defaults to $\alpha = I$, that is, newborns only feel competition from their own species.

- The function $L(x) = 0$ if $x \leq 0$, while $L(x) = x$ for $x > 0$. This is to ensure that if $C(m) > K$ no newborns are produced.
- $X(N-1, F, g_F, m_F)$ is the number of adult ($a = N-1$), females of genotype g_F and species m_F . So $\lambda X(N-1, F, g_F, m_F)$ is the number of newborns produced by these female per timestep.

- $P_{\text{mating}}(g_M, m_M, m_F)$ is the probability that a male adult of genotype g_M and species m_M successfully mates with a female adult of species m_F to produce newborn. It is

$$P_{\text{mating}}(g_M, m_M, m_F) = \frac{w(m_M, m_F)X(N-1, M, g_M, m_M)}{\sum_{g'=0}^{N_{\text{genotypes}}} \sum_{m'=0}^{N_{\text{species}}} w(m', m_F)X(N-1, M, g', m')} . \quad (7)$$

The numerator is the number of matings between male of species m_M and genotype g_M and the female, while the denominator normalises the probability. The matrix w defaults to the identity.

- $P_{\text{offspring}}(s, g, m|g_M, g_F, m_M, m_F)$ is the probability the offspring will have sex s , genotype g and species m , given the genotypes and species of its parents. This is

$$P_{\text{offspring}}(s, g, m|g_M, g_F, m_M, m_F) = h(m_M, m_F, m)i(g_M, g_F, g)p(g_M, g_F, m_M, m_F, s) . \quad (8)$$

The first term, h , determines the hybridisation between species, the second determines the inheritance of genotypes, while the final term describes any sex bias in the offspring. The hybridisation defaults to $h = 1$ if $m_M = m_F = m$ and zero otherwise.

- Finally, these expressions are summed over all possible parental genotypes and species using $\sum_{g_M, g_F, m_M, m_F}$.

3.4 Time integration

The ODEs in `Mosquito23` may be integrated in time using one of the following methods.

1. Explicit-Euler, where $X(t + \Delta t) = \Delta t f(X(t))$. This is fast, but results in the greatest error.
2. Runge-Kutta4, where $X(t + \Delta t)$ is given by the fourth-order Runge-Kutta formula. This is approximately 4 times slower than explicit-Euler.
3. Scipy's `solve_ivp` method. This is over 100 times slower than explicit-Euler, but is the most accurate.

In addition, adaptive time-stepping is the default. Here, the user defines Δt , and if the algorithm detects that any $X(t + \Delta t) < 0$, the time-step is solved using a number of smaller sub-time-steps, chosen to guarantee that all X remain non-negative. This type of behaviour occurs when the time-dependent carrying capacity suddenly reduces, and the explicit-Euler or Runge-Kutta4 methods produce large negative changes in population numbers, which, if allowed, would result in $X < 0$. Solving the problem using smaller sub-time-steps overcomes this problem. In this algorithm, there is a minimum Δt allowed, which defaults to 10^{-12} , below which the algorithm exits with an error.

Finally, a user-defined cutoff, c , is placed on $X(t + \Delta t)$. If $X(t + \Delta t) < c$ (at the end of a time step) then $X(t + \Delta t)$ is set to zero. This prevents anomalous round-off and precision errors from accumulating. The default value of c is 10^{-6} .