

21 4. Our code is based on the formulation by Nisbet and Gurney (1983) us-
22 ing delay differential equations, which are solved using the R-packages
23 `deSolve` or `PBSddesolve`.

24 1 Introduction

25 `stagePop` is an R package that can be used to model the deterministic dynamics
26 and interactions of stage-structured populations. These are populations where
27 the life cycle consists of distinct stages - e.g. eggs, juveniles and reproductive
28 adults. Explicitly including stage structure when modelling the population dy-
29 namics of stage-structured organisms can have an enormous effect on the result-
30 ing dynamics. This may be because the organism is only predated upon when it
31 is in certain life stages. Or that environmental variables, such as temperature,
32 only influence the development rate of certain stages.

33 `stagePop` has been specifically designed to investigate these sorts of eco-
34 logical problems and can therefore simulate the dynamics of stage-structured
35 populations that are involved in predator-prey interactions, host-parasitoid in-
36 teractions, resource competition, environmental change and so on. It also has
37 the ability to simulate the dynamics of any number of strains within a species
38 and therefore can be used to test questions about diversity and intra-specific
39 competition. This means it is ideally suited to investigate the timely issues of
40 biological engineering and control, biodiversity and climate change.

41 The package is based on the formulation by Bill Gurney and Roger Nisbet
42 (Nisbet and Gurney (1983); Gurney et al. (1983); Gurney and Nisbet (1998)),
43 described in detail in Appendix 1. Broadly speaking the model assumes that
44 once an individual is born, unless it dies, it moves through its different life stages
45 as if on a conveyor belt which may speed up and slow down as its development
46 rate changes. Thus, an organism begins life by being born into the first stage
47 and then, if it survives long enough, will mature into each successive stage.
48 Within each stage it is assumed that each individual has the same vital rates

49 e.g. the same death rates, the same rate of maturation etc. The formulation is
50 based on delay differential equations (which are solved within `stagePop` using
51 the R-packages `deSolve` (Soetaert et al., 2010) or `PBSddesolve` (Schnute et al.,
52 2013) and this continuous time formulation copes easily with time-varying stage
53 durations. Non-stage-structured species (which don't require delay equations)
54 may also be modelled using `stagePop` which is useful when modelling the in-
55 teractions of a number of species where not all have distinct life stages. In this
56 paper we give a description of `stagePop` and provide a number of examples
57 demonstrating how `stagePop` can be used for different modelling projects.

58 2 Running stagePop

59 To install the `stagePop` package, in R, type `install.packages('stagePop')` fol-
60 lowed by `library(stagePop)`. To run the model the function `popModel()` is
61 called. In this section we give a brief overview of this function (further details
62 are given in Appendix 2). The output from `popModel()` is a matrix which
63 contains the values of the state variables, the probabilities of surviving each
64 stage, the durations of each stage (if time-varying) and the rates of change of
65 each state variable at the times specified, via `'timeVec'`, in the input to `pop-`
66 `Model()`. Each column of the output matrix is named using the `'speciesNames'`
67 and `'stageNames'` specified in the `popModel` inputs (see Section 1.2, Appendix
68 2). Different species may be specified in any order (as long as the definition is
69 consistent) but the life stages must be referred to in the same order as they are
70 in the life cycle. Furthermore, the birth of new organisms is assumed to be into
71 the first stage only.

72 The input arguments to `popModel()` are used to completely define the
73 model system and are described in detail in Appendix 2 (section 1.1). One
74 of these inputs is a list containing all the rate functions (e.g. death rates,
75 reproduction rates etc) for all entities in the model. Appendix 2 (section 1.1.1)
76 gives a detailed description on how these functions must be defined.

77 Also included in the `stagePop` package are the following functions:

- 78 • `checkSolution()` produces warnings if the solution contains any negative
79 values
- 80 • `genericPlot()` provides a basic plot of the results (most of the figures in
81 this paper have been generated automatically by `stagePop`)
- 82 • `plotStrains()` if there are multiple strains in a species, will plot them
83 individually (`genericPlot()` only plots the sum of the strains).
- 84 • `runStagePopExample` runs the examples shown in section 3, e.g. run-
85 `StagePopExample('BlowFlies')`
- 86 • `sumStrains` if there are multiple strains in a species, sums the model
87 output over the strains in each species.

88 These functions are automatically called in the `popModel()` function, un-
89 less the user specifies otherwise, but can also be used on a stand alone basis.
90 In Appendix 2 we give some tips on how to check the solution generated from
91 `stagePop` is accurate (Appendix 2, section 3) and some ideas on trouble shoot-
92 ing typical problems that may occur with the delay differential equation solvers
93 (Appendix 2, section 4).

94 3 Example Applications

95 In this section we demonstrate how `stagePop` can be used to simulate a wide
96 range of problems involving stage-structured populations. Where possible, in
97 order to verify our software is working correctly, we have reproduced published
98 examples. The scripts for all of these examples are included in the `stagePop`
99 package¹, are reproduced in Appendix 3 and are also attached as supplementary
100 files. They are intended to serve as a template for users when defining their own
101 problems. The name of the appropriate script is given in square brackets in each

¹The location of these files can be found by `'system.file("DemoFiles/ExampleFileName.R",
package = "stagePop")'`

102 example heading and they can be run in R (after (`library(stagePop)`')) using
103 `runStagePopExample('BlowFlies')` (for the BlowFlies.R example).

104 **3.1 Single Species with fixed death rates and stage dura-** 105 **tions [BlowFlies.R]**

106 A classic example of a stage-structured population is Nicholson's Blowflies (Gur-
107 ney et al. (1983); Nicholson (1954, 1957)). Australian sheep blowflies, which
108 have five distinct developmental stages, grown under controlled conditions in a
109 laboratory experiment were found to exhibit sustained, large, quasi-cyclic fluc-
110 tuations in their adult populations. To reproduce these experiments, the per
111 capita death rates and duration of each stage are assumed to be constant (values
112 are given in the caption for Fig. 1 and in Script 1, Appendix 3) and the repro-
113 ductive rate (i.e. rate of egg production) is defined as $8.5 \exp\left(\frac{-A(t)}{600}\right)$ where
114 the $A(t)$ is the reproductive adult stage (stage 5) and eggs are stage 1. The
115 simulation is initiated with the immigration of 100 adults per day over the first
116 day (this is fairly arbitrary however - the magnitude of the rate of immigration
117 does not affect the equilibrium state results; similarly the immigration can be
118 into any life stage). Script 1 (Appendix 3) shows how `popModel` can be used
119 to simulate this situation. Fig 1 shows the plot automatically generated by
120 `stagePop` (compare with Fig. 3 by Gurney et al. (1983)).

121 **3.2 A single species with density-dependent death rates:** 122 **Larval Competition [LarvalComp.R]**

123 In this second example from Gurney et al. (1983), a two stage moth population
124 (larvae and adults) is considered in which larval competition for resources re-
125 sults in a density dependent per capita larval death rate given by $\alpha L(t)$ where
126 $\alpha=5 \times 10^{-5}$ moths⁻¹ d⁻¹ and $L(t)$ is the density of larvae at time t . The larval
127 stage duration is 28 d and reproduction (by adults) is given by $qA(t)$ where
128 $q = 9.4$ eggs/adult/d and $A(t)$ is the density of adults at time t . Two different

129 cases are considered; in the first, the adult death rate is fixed at 0.2 d^{-1} , in the
 130 second, the adults are assumed to die after 5 days. In this second case the adult
 131 death rate is set to zero and a third stage (corpses) is added to the model. We
 132 begin both of the simulations with the immigration of adults at a rate of 20 d^{-1}
 133 over the first day. The results are shown in Fig. 2 and Figs. 2c and 2d can be
 134 compared with Fig. 4 by Gurney et al. (1983). This example demonstrates the
 135 huge difference in population dynamics caused by different ways of modelling
 136 the death of adults. The code required to run either of these cases in `stagePop`
 137 is shown in Script 2, Appendix 3.

138 **3.3 A single species whose stage durations depend on tem-** 139 **perature [VarDurEnv.R]**

140 Unfortunately we could not find a sufficiently simple published example of a
 141 continuous-time model of a stage-structured population affected by temperature
 142 change (although there are more complex ones, e.g. Beck-Johnson et al. (2013)),
 143 so we have formulated our own example.

We consider a theoretical species with two stages (juvenile and adult) where
 growth experiments conducted over a range of different but constant temper-
 atures, T_c , have shown that the length of the juvenile stage, τ , is affected by
 temperature according to:

$$\tau(T_c) = \tau_{\min} + \left(\frac{T_c - T_{opt}}{w} \right)^2 \quad (1)$$

where $T_{opt} = 20^\circ\text{C}$, $w = 2^\circ\text{C d}$ and $\tau_{\min} = 60 \text{ d}$ (see Fig. 3a and `tauFunc` in
 Script 3 in Appendix 3). We use this relationship to define the instantaneous
 juvenile development rate,

$$g(T(t)) = \frac{1}{\tau(T(t))} \quad (2)$$

144 which we assume will also apply to time-varying temperatures, $T(t)$. To inform

145 **stagePop** that we are dealing with a problem involving a time-varying stage
 146 duration we set ‘**timeDependDuration**’ equal to TRUE. Since in this example
 147 τ is now changing with time, we define **develFunc** using Eq. 2 and this is
 148 used to compute the rate of change of $\tau(T(t))$ (Eq. 9 in Appendix 1. The
 149 **durationFunc** (which is used to define non-time-varying durations) is now only
 150 used to define the length of the stage duration at the beginning of the simulation
 151 (see Script 3, Appendix 3).

We then simulate the growth of the species over a number of years where the temperature, T , varies over an annual cycle according to,

$$T(t) = T_a(1 - \cos(2\pi(t + 80)/365)) \quad (3)$$

152 where the yearly average temperature, T_a , is 15° C, t is in days and the time
 153 offset of 80 d is required to prevent the species dying out due to low temperatures
 154 when the population is small at the start of the simulation. This is defined in
 155 the function **tempFunc** in Script 3 (Appendix 3) and displayed in Fig. 3c.

156 The simulation begins with the immigration of adults and reproduction is
 157 assumed to be density dependent. The definition of this model is described for
 158 **stagePop** using Script 3 (Appendix 3) and the results are shown in Fig. 3. The
 159 changes in the juvenile stage duration τ over time, computed from **stagePop**
 160 are shown in Fig. 3d and these are compared with the value of τ computed from
 161 Eq. 1 which is the stage duration if the current temperature, $T(t)$, had been
 162 constant over the stage duration.

163 3.4 Two interacting species: Predator-Prey System

164 [PredPrey1.R] and [PredPrey2.R]

165 In this example we show how **stagePop** can be used to model two species –
 166 a predator and its prey. We begin with the classic Lotka-Volterra predator-
 167 prey model where neither species has stage structure (PredPrey1.R) and then
 168 increase the complexity by adding in stage structure for the predator and then

169 density dependent death for the prey (PredPrey2.R) which is the system studied
 170 by Gourley and Kuang (2004).

The classic Lotka-Volterra equations (Lotka, 1925) are given by

$$\dot{x}(t) = rx(t) - py(t)x(t) \quad (4)$$

$$\dot{y}(t) = bpy(t)x(t) - Dy(t) \quad (5)$$

171 where $x(t)$ and $y(t)$ are the prey and predator densities at time t respectively,
 172 $rx(t)$ is the prey reproduction rate, $py(t)$ is the per capita death rate of prey
 173 due to predation, $bpy(t)x(t)$ is the predator reproduction rate and D is the per
 174 capita death rate of the predator.

175 This system is defined in `stagePop` as shown in Script 4 (Appendix 3).
 176 The results of the simulation are shown in Fig. 4. The analytical solution
 177 to Equations 4 and 5 is the closed loop shown in Fig. 4b. However, if the
 178 tolerances on the DDE solver are not strict enough, the solution will be subject
 179 to numerical errors and the predator-prey loop in Fig. 4b will not be closed. For
 180 example changing the value of `'tol'` in `'solverOptions'` from `'1e-7'` to `'1e-3'`
 181 gives the result shown in Figs. 4c and 4d (see Appendix 2, Section 3, for further
 182 tips on how to check your solution is accurate).

We now look at the case where the predator has juvenile and adult stages (y_j
 and y respectively) and only the adult stage consumes the prey. The equations
 now become

$$\dot{x}(t) = rx(t) - py(t)x(t) \quad (6)$$

$$\dot{y}_j(t) = bpy(t)x(t) - bpy(t - \tau_j)x(t - \tau_j)\exp(-D_j\tau_j) - D_jy_j(t) \quad (7)$$

$$\dot{y}(t) = bpy(t - \tau_j)x(t - \tau_j)\exp(-D_j\tau_j) - Dy(t). \quad (8)$$

183 where D_j is the juvenile predator per capita death rate (here set at 1 d^{-1}) and
 184 τ_j is the length of the juvenile predator stage duration. The `stagePop` code for
 185 this new situation is in Script 5 (Appendix 3) (where `'case=1'`) and the results

186 of the simulation for $\tau_j=0.1$ are shown in Fig. 5. It is clear that adding in
 187 a juvenile stage which does not predate causes large changes compared to the
 188 equilibrium situation shown in Fig. 4 (a and b).

We now add in a density dependent death rate for the prey such that the prey equation becomes

$$\dot{x}(t) = rx(t)\left(1 - \frac{x(t)}{K}\right) - py(t)x(t) \quad (9)$$

189 which is the system investigated by Gourley and Kuang (2004) (Gourley and
 190 Kuang, 2004). To run this in `stagePop` we only need state a value for K and
 191 modify `deathFunc` as shown for cases >1 in Script 5 (Appendix 3) With $K=1$
 192 and the other parameters as before, Fig. 6 shows the results when the predator
 193 juvenile stage duration is 0.1 d and 1.8 d (cases 2 and 3 respectively in Script
 194 5 (Appendix 3). Note the length of the simulation has been increased so that
 195 these plots can be more easily compared with results in Gourley and Kuang
 196 (2004). The results from `stagePop` compare well with these until $\tau_j \geq 15$ after
 197 which there are large, unaccounted for discrepancies between the simulations
 198 (cases 6 and 7 in `PredPrey2.R` (Script 5, Appendix 3)).

199 3.5 Multiple interacting species: Host-Parasitoid System

200 [Briggs.R]

201 This example considers three interacting species, all with stage structure, in the
 202 host-parasitoid system investigated by Briggs (1993). The host species has 3
 203 life stages (eggs, E ; larvae, L and adult, A) and is attacked by two competing
 204 parasitoids: P , which attacks the host eggs and, Q , which attacks the host
 205 larvae. Both parasitoids have 2 life stages – juvenile (P_J and Q_J) and adult
 206 (P_A and Q_A). The egg and larval attack rates are denoted by a_P and a_Q
 207 respectively. Each parasitised host becomes a single juvenile parasitoid, thus
 208 the death rates due to parasitoids are $a_P P_A(t)E(t)$ and $a_Q Q_A(t)L(t)$ for the
 209 host eggs and larvae respectively, and reproduction into the parasitoid juvenile

210 class is $a_P E(t)P_A(t)$ for P and $a_Q L(t)Q_A(t)$ for Q . Reproduction of the host
 211 is given by $\rho D_A A(t)$ where ρ is host lifetime fecundity and D_A is the adult
 212 death rate (see Briggs (1993) for the equations describing this system). In this
 213 example we use the steady solution (provided in Appendix B by Briggs (1993))
 214 to set the values for the immigration rates (these parameters are identified by
 215 the ‘star’ in their name - e.g. Q_{star}). The `stagePop` code for this system is
 216 shown in Script 6 (Appendix 3).

217 Theory dictates (Briggs, 1993) that in this situation the two parasitoids can
 218 not co-exist. We use `stagePop` to simulate an invasion of P (after 20 time units)
 219 into a situation where only Q and the host are initially present. For the case
 220 in which the parasitoid Q has twice the attack rate of P ($a_P=1$, $a_Q=2$) and
 221 all their other parameters are identical, P still manages to displace Q since it
 222 attacks at an earlier stage (eggs rather than larvae) and the system settles at a
 223 new equilibrium with a higher adult host density (Fig. 7). These simulations
 224 are interesting from the point of view of biological control - if the adult host is a
 225 pest which causes damage e.g. to people or crops, the most desirable parasitoid
 226 to release is that which minimises the adult host density when in equilibrium.

227 3.6 Consumer-Resource problem with variable stage du- 228 ration [VarDurFood.R]

Here we use `stagePop` to reproduce an example given by Nisbet and Gurney
 (1983) in which the length of the larval stage of a 2-stage species (loosely based
 on the damselfly) is determined by the availability of their food. Specifically, an
 individual larva becomes an adult once it has assimilated enough food to raise
 its body mass by m mass units. Thus, by definition the larval stage duration,
 $\tau_L(t)$ is determined by the equation

$$\int_{t-\tau_L(t)}^t g_L(x) dx = m, \quad (10)$$

where $g_L(t)$ is the larval development rate (see Appendix 1, Eq. 7). It is assumed that $g_L(t)$, is proportional to the rate of food consumption per larva, $f_L(t)$, such that

$$g_L(t) = \epsilon f_L(t), \quad (11)$$

and

$$f_L(t) = f_{\max} \frac{F(t)}{K + F(t)}, \quad (12)$$

229 where $F(t)$ is the food density, K is the half saturation constant and f_{\max} is
 230 the maximum food consumption rate. Thus the rate of food uptake by larvae
 231 is $f_L L(t)$. Food is supplied to the larvae at a constant rate f_s and the adults
 232 have a fixed rate of reproduction, $qA(t)$ where $A(t)$ is the adult density at time
 233 t . Both larvae and adults are assumed to have fixed per capita death rates D_L
 234 and D_A respectively.

To solve this problem in **stagePop** we define it as a two species problem where food is one species and the damselfly is the other. The ‘reproduction’ of food is the constant rate of food supply, f_s , and its ‘death’ is modelled by the per capita rate uptake rate, $f_L L(t)/F(t)$, i.e.

$$\frac{f_{\max}}{K + F(t)} L(t). \quad (13)$$

A time dependent death rate is specified for the food species and a time dependent duration for the ‘damselfly’ using the **popModel()** arguments ‘**timeDependLoss**’ and ‘**timeDependDuration**’ respectively. Since the length of the stage duration is changing in time, the **durationFunc** is only required at $t=0$, and the development rate is set in **develFunc** using Eqs. 11 and 12. To set the initial value of τ_L it is assumed that for $t < 0$ the development rate is constant, thus Eq. 10 implies $g_L(0)\tau_L(0) = m$. At the beginning of the simulation the larvae have an initial amount of food, $F(0)$ and thus $g_L(0)$ can be computed from Eqs. 11 and 12 to give

$$\tau_L(0) = m \frac{(K + F(0))}{\epsilon f_{\max} F(0)}. \quad (14)$$

235 The instructions for `stagePop` are given in Script 7 (Appendix 3) and the results
 236 are shown in Fig. 8.

237 This example shows the flexibility of using the Nisbet-Gurney formulation –
 238 the stage duration can be controlled by any model variable, allowing size, age,
 239 weight etc to determine the time of transition into the next life stage.

240 3.7 Consumer-Resource model with multiple strains in 241 one species [MultipleStrains.R]

In this example we demonstrate how a species with multiple strains can be modelled in `stagePop`. We begin with looking at a simplified model of bacteria in the human colon. The bacteria feed on a resource R (e.g. food that has not been digested further up the gut) and are subject to transport through the gut at a rate of V . Assuming Monod-equation type growth, the rate of change in concentration (or density) of bacterial strain i , is given by

$$\frac{dB^i(t)}{dt} = G^i \frac{R(t)}{R(t) + K} B^i(t) - V B^i(t) \quad (15)$$

(e.g. Kettle et al. (2014)) where G^i is the maximum specific growth rate of strain i and K is the half saturation constant (assumed constant over all strains). The rate of change of resource is given by

$$\frac{dR(t)}{dt} = V R_{in} - \frac{R(t)}{R(t) + K} \sum_{i=1}^N \frac{G^i B^i(t)}{Y} - V R(t) \quad (16)$$

242 where N is the number of strains, Y is the yield (i.e. the number of grams of
 243 B produced from one gram of R) and R_{in} is the concentration of the incoming
 244 resource. This is modelled in `stagePop` as two species – species one is the
 245 resource and species 2 is the bacteria (Script 8, Appendix 3). With no stage
 246 structure the system rapidly reaches steady state with one strain dominating
 247 the system (competitive exclusion; Fig. 9a and b).

However, bacteria may have a lag phase during which time there is little or

no cell growth but the cells are busy replicating various proteins and DNA in preparation for the reproductive phase. For demonstration purposes, we assume the length of this phase, τ_i , varies slightly between strains, and whilst in this stage the bacteria are not subject to the usual transport through the system. Thus for the lagged stage, B_1 ,

$$\frac{dB_1^i(t)}{dt} = G^i \frac{R(t)}{R(t) + K} B_2^i(t) - m^i(t) \quad (17)$$

where $m^i(t)$ is the maturation rate of strain i from stage one at time t ; and for the reproductive stage, B_2 ,

$$\frac{dB_2^i(t)}{dt} = m^i(t) - V B_2^i(t). \quad (18)$$

The rate of change of resource is now given by

$$\frac{dR(t)}{dt} = V R_{in} - \frac{R(t)}{R(t) + K} \sum_{i=1}^N \frac{G^i B_2^i(t)}{Y} - V R(t). \quad (19)$$

248 When assigning the strain traits (G^m and τ_i), we assume a trade-off such that a
 249 longer lag time leads to faster growth. We incorporate this second case (Script
 250 8, Appendix 3) and see that this has a significant effect on the results (Fig.
 251 9c-e). The system now does not reach steady state even over the extended time
 252 period shown and multiple strains are still co-existing after 100 time units.

253 4 Conclusion

254 The preceding sections demonstrate only a small range of the possible problems
 255 **stagePop** can be used to investigate. However, we hope that these demonstrate
 256 its flexibility and potential, and that other researchers will find **stagePop** useful
 257 in their own fields.

258 Supporting Information

- 259 • **Appendix 1** Mathematical Formulation
- 260 • **Appendix 2** Detailed descriptions of `stagePop` functions
- 261 • **Appendix 3** PDF of R scripts for the Example Applications (`Blowflies.R`,
262 `LarvalComp.R`, `VarDurEnv.R`, `PredPrey1.R`, `PredPrey2.R`, `Briggs.R`, `VarDurFood.R`,
263 `MultipleStrains.R`)

264 Acknowledgements

265 This work was financially supported by the German Federal Ministry for Ed-
266 ucation and Research (research grant no. FKZ01LL0917A-01LL0917O, for the
267 LEGATO project) and the Scottish Government's Rural and Environment Sci-
268 ence and Analytical Services Division (RESAS).

269 References

- 270 Beck-Johnson, L., Nelson, W., Paaijmans, K., Read, A., Thomas, M., and
271 Bjørnstad, O. (2013). The effect of temperature on anopheles mosquito pop-
272 ulation dynamics and the potential for malaria transmission. *PLOS one*,
273 8:79276.
- 274 Briggs, C. (1993). Competition among parasitoid species on a stage-structured
275 host and its effect on host suppression. *The American Naturalist*, 141:372–397.
- 276 Gourley, S. and Kuang, Y. (2004). A stage structured predator-prey model and
277 its dependence on maturation delay and death rate. *Mathematical Biology*,
278 49:188–200.
- 279 Gurney, W. and Nisbet, R. (1998). *Ecological Dynamics*. Oxford University
280 Press.

- 281 Gurney, W., Nisbet, R., and Lawton, J. (1983). The systematic formulation of
282 tractable single-species population models incorporating age structure. *Jour-*
283 *nal of Animal Ecology*, 52:479–495.
- 284 Kettle, H., Donnelly, R., Flint, H., and G., M. (2014). ph feedback and pheno-
285 typic diversity within bacterial functional groups of the human gut. *Journal*
286 *of Theoretical Biology*, 342:62–69.
- 287 Lotka, A. (1925). *Elements of Physical Biology*. Williams and Wilkins.
- 288 Nicholson, A. (1954). An outline of the dynamics of animal populations. *Aus-*
289 *tralian Journal of Zoology*, 2:9–65.
- 290 Nicholson, A. (1957). The self-adjustment of populations to change. *Cold spring*
291 *Harbour symposium on Quantitative Biology*, 22:153–173.
- 292 Nisbet, R. and Gurney, W. (1983). The systematic formulation of population
293 models for insects with dynamically varying instar duration. *Theoretical Pop-*
294 *ulation Biology*, 23:114–135.
- 295 Schnute, J., Couture-Beil, A., Haigh, R., and Kronlund, A. (2013). PBSmod-
296 elling R package. <http://code.google.com/p/pbs-modelling/>.
- 297 Soetaert, K., Petzoldt, T., and Woodrow Setzer, R. (2010). Solving differential
298 equations in r: Package desolve. *Journal of Statistical Software*, 33(9):1–25.
299 <http://www.jstatsoft.org/v33/i09/>.

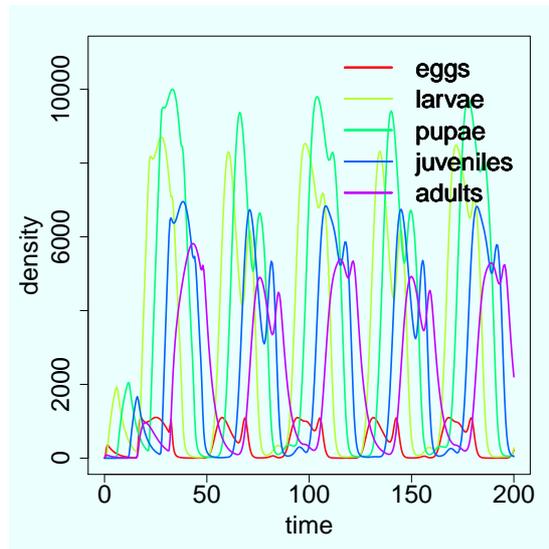


Figure 1: Simulation of the five life stages of Nicholson's blow flies (Section 3.1). The per capita death rates for stages 1-5 are 0.07, 0.004, 0.003, 0.0025, 0.27 d^{-1} and the durations of stages 1-4 are 0.6, 5.0, 5.9, 4.1 d. Compare with Fig. 3 by Gurney et al. (1983).

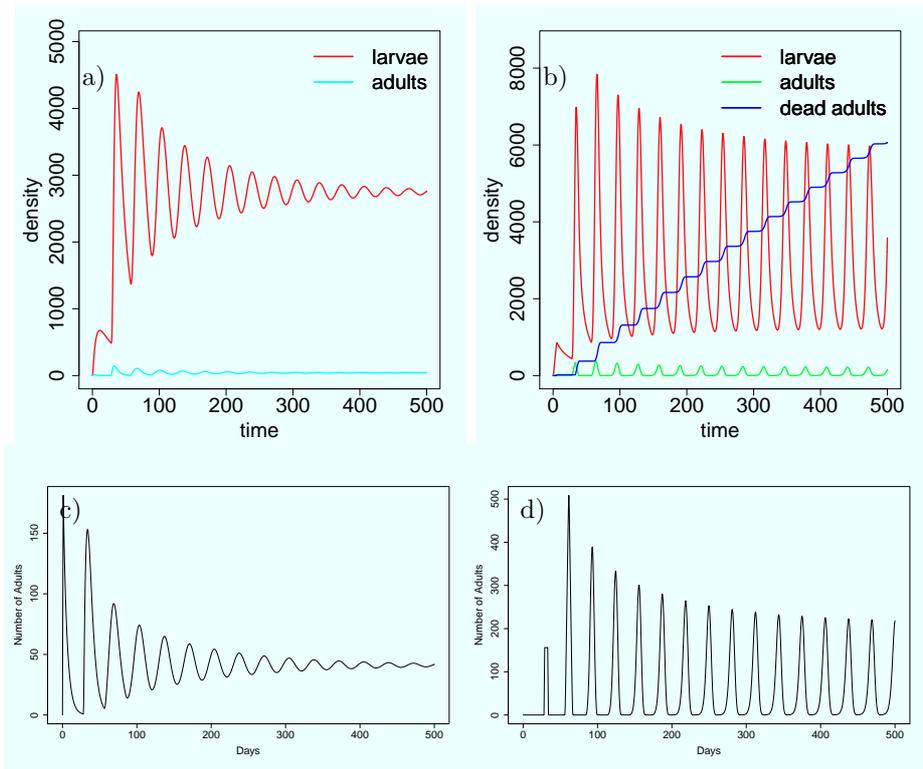


Figure 2: Modelling larval competition. a) the adult death rate is constant at 0.2 d^{-1} ; b) the adults have a fixed lifetime of 5 d; c) and d) are the same as a) and b) but show only adults for comparison with Fig. 4 a and b, by Gurney et al. (1983).

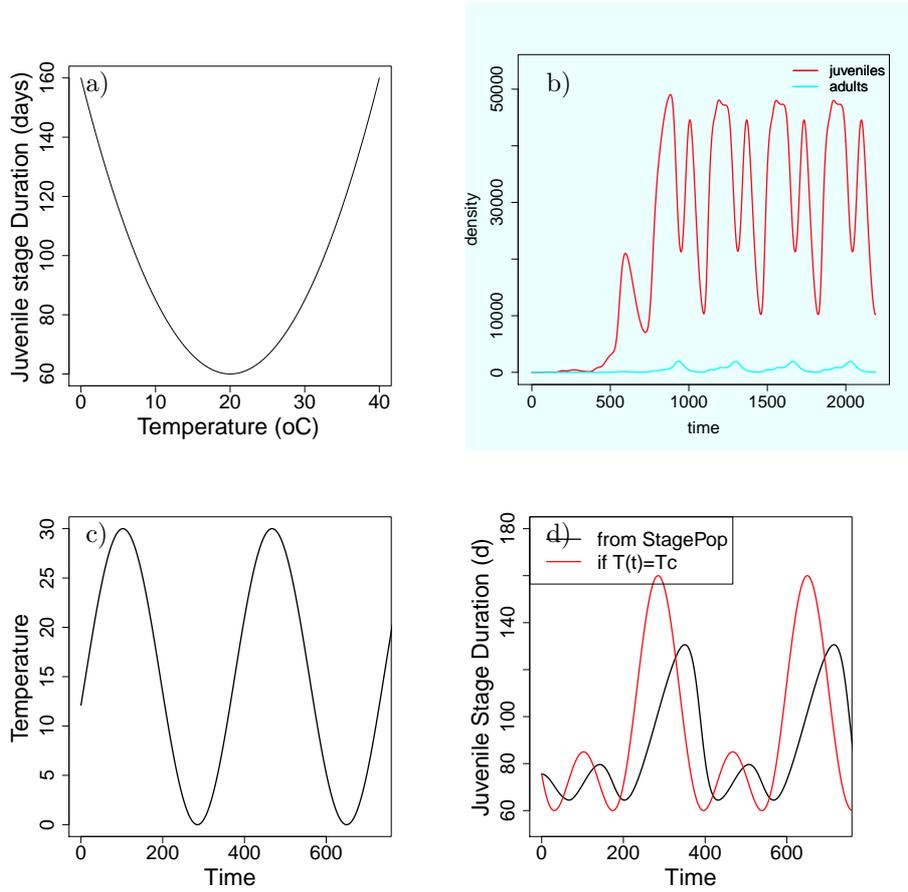


Figure 3: Simulation of a single species with a juvenile and adult stage where the juvenile development rate is temperature dependent (Section 3.3). a) Relationship between juvenile stage duration and temperature (Section 3.3 (Eq. 1)); b) Results from `stagePop`; c) temperature time series; d) comparison of calculations of τ . Note the lower two plots are shown for a shorter time period (2 years rather than 6 years) for clarity.

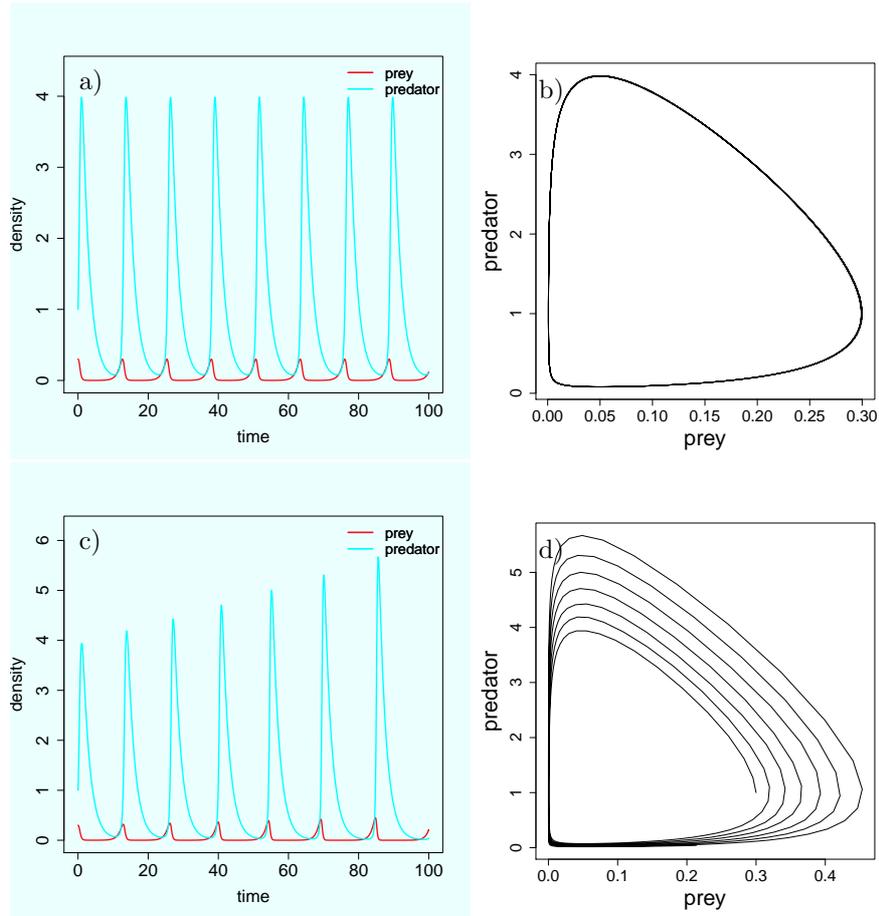


Figure 4: a) Predator-prey dynamics with no stage structure. The closed loop in b) indicates the solution is numerically accurate ('tol'=1e-7 in 'solverOptions'); d) when 'tol'=1e-3 in 'solverOptions', numerical accuracy has not been achieved (and plot c) is incorrect) as the predator-prey graph is no longer a closed loop.

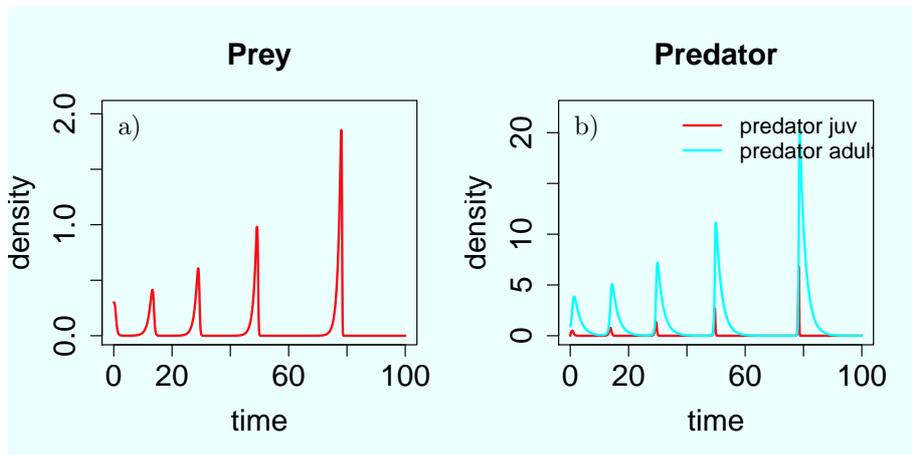


Figure 5: Predator-prey dynamics where the predator has a juvenile stage of duration 0.1 d.

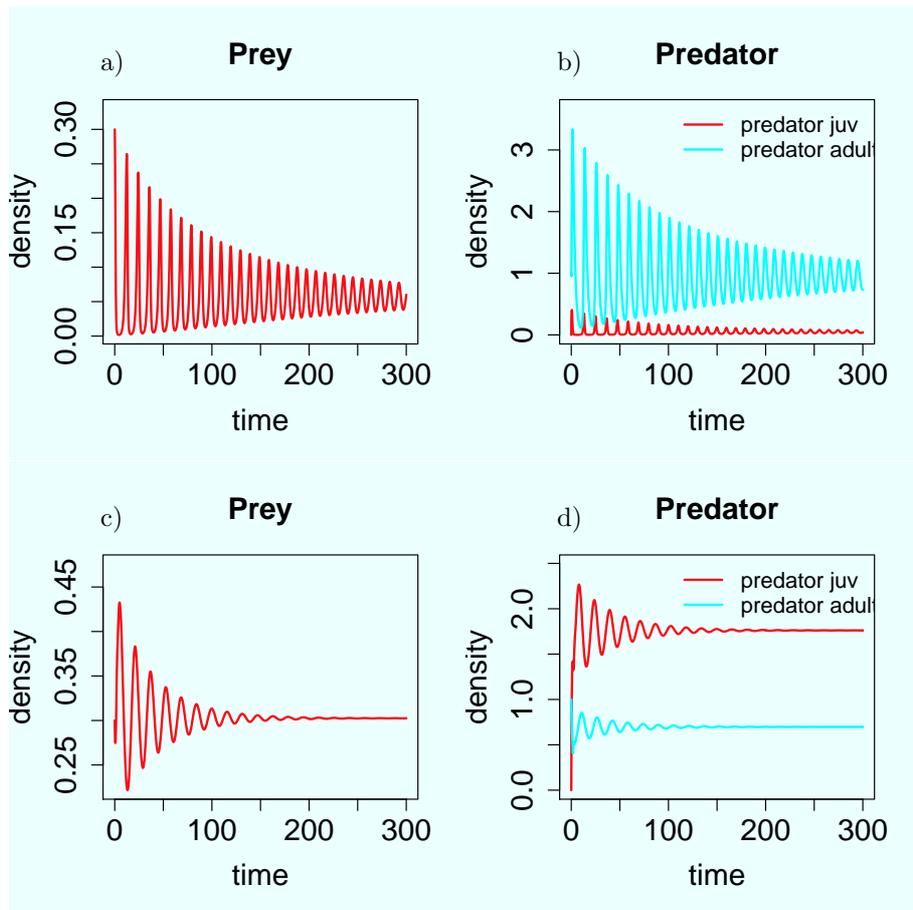


Figure 6: Predator-prey dynamic where the prey has density dependent death and the predator has a juvenile stage of duration 0.1 d (top row) and 1.8 d (bottom row). Compare with Fig. 3 by Gourley and Kuang (2004).

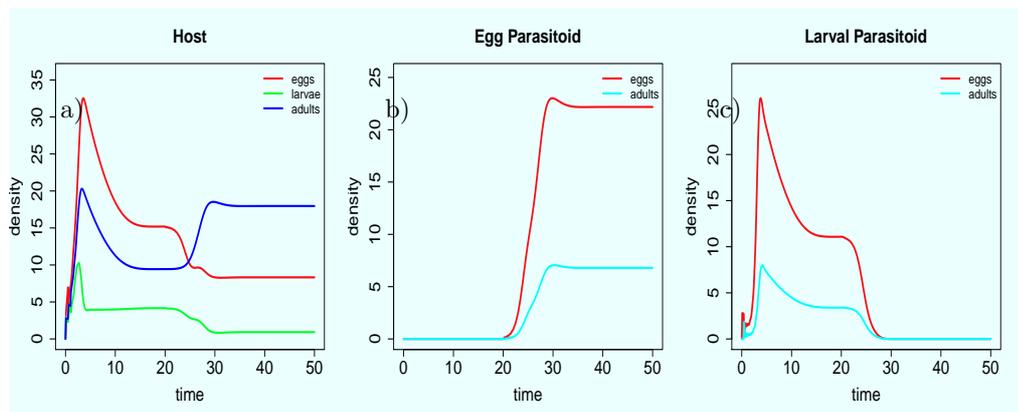


Figure 7: Competition between two parasitoids attacking different life stages of one host (compare with Fig.3 (Briggs, 1993)). See Script 6 (Appendix 3) for parameter values.

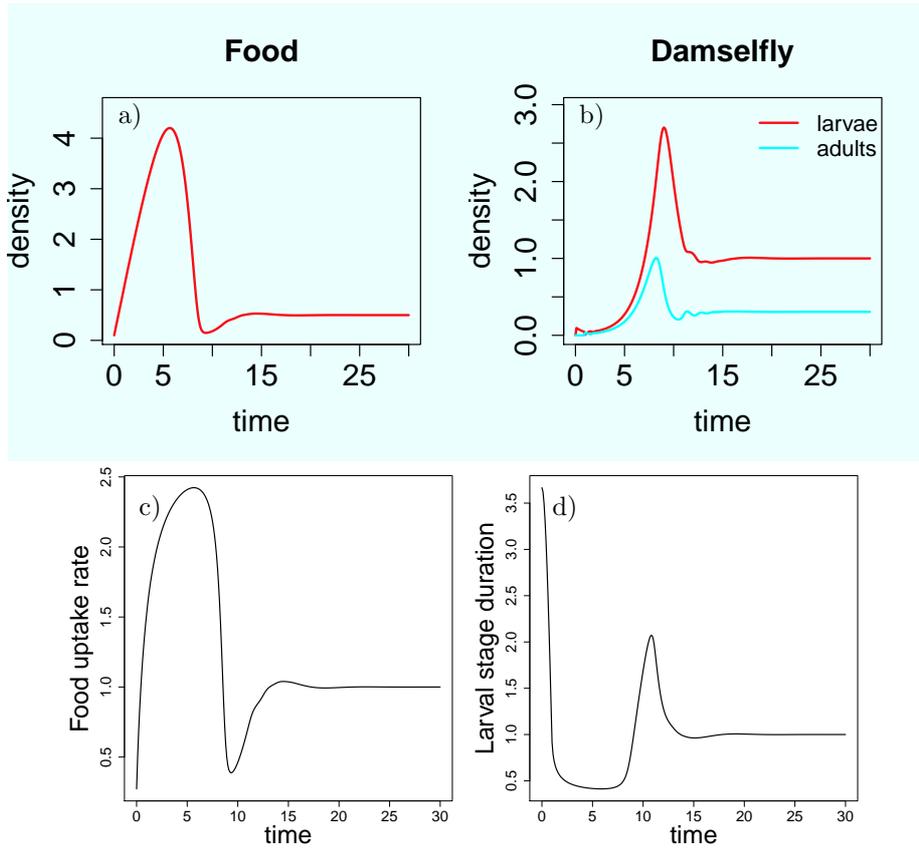


Figure 8: Example where length of larval maturation time is determined by food availability (Section 3.6). The simulation begins with $L(0) = A(0) = 0$, $F(0) = 0.1$ with immigration into the larval stage at rate 1 per unit time for the first 0.1 time units. The parameters used to achieve these plots (which compare well with Fig. 3 (Nisbet and Gurney, 1983)) are $f_s = m = \epsilon = K = 1$, $D_A=2$, $q=5$, $D_L = \ln(\frac{q}{D_A})$ and $f_{\max} = 3$.

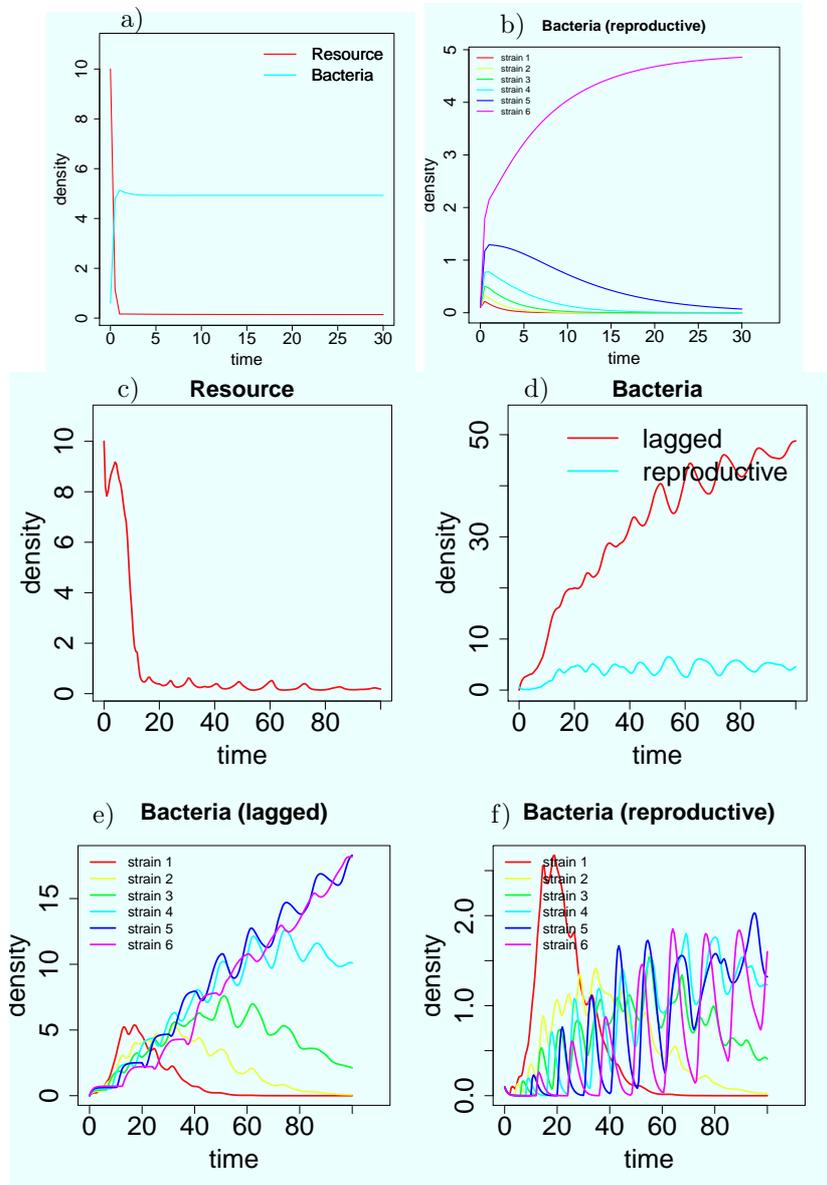


Figure 9: Example using multiple strains in one species. Model with no stage structure: a) where ‘Bacteria’ is the sum over all strains, and b) time evolution of the 6 individual strains. Model with stage-structure: Time evolution of resource (c), the two stages of bacteria (d), and the individual strains for the lagged stage (e) and the reproductive stage (f). Note extended time period for the stage-structured model.