ECOGRAPHY

Software notes

RangeShiftR: an R package for individual-based simulation of spatial eco-evolutionary dynamics and species' responses to environmental changes

Anne-Kathleen Malchow, Greta Bocedi, Stephen C. F. Palmer, Justin M. J. Travis and Damaris Zurell

A. Malchow (https://orcid.org/0000-0003-1446-6365) ✉ (malchow@uni-potsdam.de) and D. Zurell (https://orcid.org/0000-0002-4628-3558), Univ. of Potsdam, Inst. of Biochemistry and Biology, Potsdam, Germany; and Humboldt Univ. of Berlin, Geography Dept, Berlin, Germany. – G. Bocedi (https://orcid.org/0000-0002-9131-6670), S. C. F. Palmer and J. M. J. Travis (https://orcid.org/0000-0002-5785-4272), School of Biological Sciences, Univ. of Aberdeen, Zoology Building, Tillydrone Avenue, Aberdeen, UK.

Reliably modelling the demographic and distributional responses of a species to environmental changes can be crucial for successful conservation and management planning. Process-based models have the potential to achieve this goal, but so far they remain underused for predictions of species' distributions. Individual-based models offer the additional capability to model inter-individual variation and evolutionary dynamics and thus capture adaptive responses to environmental change.

We present RangeShiftR, an R implementation of a flexible individual-based modelling platform which simulates eco-evolutionary dynamics in a spatially explicit way. The package provides flexible and fast simulations by making the software RangeShifter available for the widely used statistical programming platform R. The package features additional auxiliary functions to support model specification and analysis of results. We provide an outline of the package's functionality, describe the underlying model structure with its main components and present a short example.

RangeShiftR offers substantial model complexity, especially for the demographic and dispersal processes. It comes with elaborate tutorials and comprehensive documentation to facilitate learning the software and provide help at all levels. As the core code is implemented in C++, the computations are fast. The complete source code is published under a public licence, making adaptations and contributions feasible.

The RangeShiftR package facilitates the application of individual-based and mechanistic modelling to eco-evolutionary questions by operating a flexible and powerful simulation model from R. It allows effortless interoperation with existing packages to create streamlined workflows that can include data preparation, integrated model specification and results analysis. Moreover, the implementation in R strengthens the potential for coupling RangeShiftR with other models.

Keywords: connectivity, conservation, dispersal, evolution, population dynamics, range dynamics

www.ecography.org

Introduction

Under anthropogenic exploitation and rapid environmental changes, one of the most urgent challenges biologists face today is to understand and predict if and how species will persist, by adapting or undergoing changes in their geographic range (McGill et al. 2015, Brondizio et al. 2019). To infer a species' niche from data and make predictions in space and time, correlative species distribution models (SDMs) are commonly used tools (Guisan and Zimmermann 2000, Elith and Leathwick 2009, Qiao et al. 2015). The widespread use of SDMs has been facilitated by accessible and ready-to-use software, most notably Maxent (Phillips et al. 2017) and dedicated R packages such as biomod2 (Thuiller et al. 2009) and dismo (Hijmans et al. 2017). However, these methods often incorporate little ecological theory (Guisan and Thuiller 2005, Austin 2007) and usually require making assumptions that are routinely violated in natural observed systems (Elith et al. 2010, Jarnevich et al. 2015, Martínez-Minaya et al. 2018). For example, SDMs assume that species are at equilibrium with their environment and ignore any transient dynamics (Zurell et al. 2016). An alternative that avoids some of these drawbacks is the development and application of mechanistic models, which aim to simulate relevant eco-evolutionary processes such as dispersal, demography and evolution (Urban et al. 2016, Cabral et al. 2017). Despite repeated calls for more mechanistic understanding of range dynamics (Kearney and Porter 2009, Schurr et al. 2012, Connolly et al. 2017), such models remain underused, arguably due to challenges such as poor availability of the data needed for parametrisation and restricted accessibility to the software required to run them (Dormann et al. 2012, Briscoe et al. 2019).

The ambition for a more prominent representation of process-based models in ecological research led to the development of the standalone software RangeShifter (Bocedi et al. 2014), a flexible individual-based model (IBM) that simulates spatial eco-evolutionary dynamics for a given species. It models population dynamics, dispersal and evolution as interacting processes, organised within a modular structure in which each process has a number of modelling options. This makes RangeShifter a highly adaptable platform with a wide range of applications, including conducting population viability or connectivity analyses (Aben et al. 2016, Henry et al. 2017) and assessing the dynamics of genetic variation across complex landscapes. The new RangeShifter ver. 2.0 (Bocedi et al. 2021) adds novel features including the option for dynamic landscapes and a completely revised genetics module. Here, we present RangeShiftR ver. 1.0, a package that implements the RangeShifter 2.0 simulation in R (<www.r-project.org>), making it multi-platform software.

With the RangeShiftR package, we take a step towards more accessible and integrated use of mechanistic individual-based models. RangeShiftR extends the existing suite of R packages for ecological modelling, which includes software like the spatially explicit population models steps (Visintin et al. 2020) and demoniche (Nenzén et al. 2012),

by a complete and flexible IBM with detailed dispersal dynamics, thus expanding the range of representable ecological levels from the population to the individual. The package augments the RangeShifter platform with functionality to assist in model specification and output visualisation. As part of the R environment, RangeShiftR offers the powerful potential to interoperate with other packages in order to form integrated workflows, drawing on the extensive functionality for data preparation, output analysis and easy reporting that is available for R. RangeShiftR is published under the public licence GPLv3 and hence may be used, modified and shared under the terms of the GPLv3. In order to provide easy access for all users, the package includes extensive built-in documentation and comes with elaborate tutorials presented on the accompanying website (<https://rangeshifter.github.io/RangeshiftR-tutorials/>).

Package structure and implementation

The RangeShiftR package inherits its model structure from the underlying RangeShifter platform (Fig. 1). It models the abundance and distribution of a population of a single species by explicitly and stochastically simulating three main interacting processes – demography, dispersal and evolution (genetics) – at the individual level. The simulation is based on a regularly gridded landscape and runs over discrete yearly or seasonal time steps. Various levels of output can be written to text files at specified time intervals during the simulation, recording data including abundance, individual traits, connectivity between patches and dispersal paths.

To reflect this conceptual structure, the RangeShiftR package contains a suite of functions and classes (Fig. 2), comprising three groups: model functions to set up the simulation, helper functions to assist with parameter specification, and output functions to process and visualise the simulation output. The helper and output functions are provided to enhance usability and constitute unique functionality of the R package not available in RangeShifter.

Model functions

A RangeShiftR model is defined by the assembly of various modules, each of which is represented in R by its own class. The model functions are their corresponding class constructors: They are used to create objects that hold the given (numeric) values of all model parameters relevant to the respective module. The species model, i.e. the part of the model that describes the study species, comprises three modules that represent the distinct simulated processes and are constructed by *Demography()*, *Dispersal()* and the optional *Genetics()*. Modules may have optional or obligatory sub-modules: demography can have a *StageStructure()*, and dispersal always comprises three phases (Travis et al. 2012), namely *Emigration()*, one of the '*Transfer*' sub-modules and *Settlement()*. The '*Transfer*' class is implemented as a virtual class that can take the form of one of the three possible sub-modules *DispersalKernel()*,
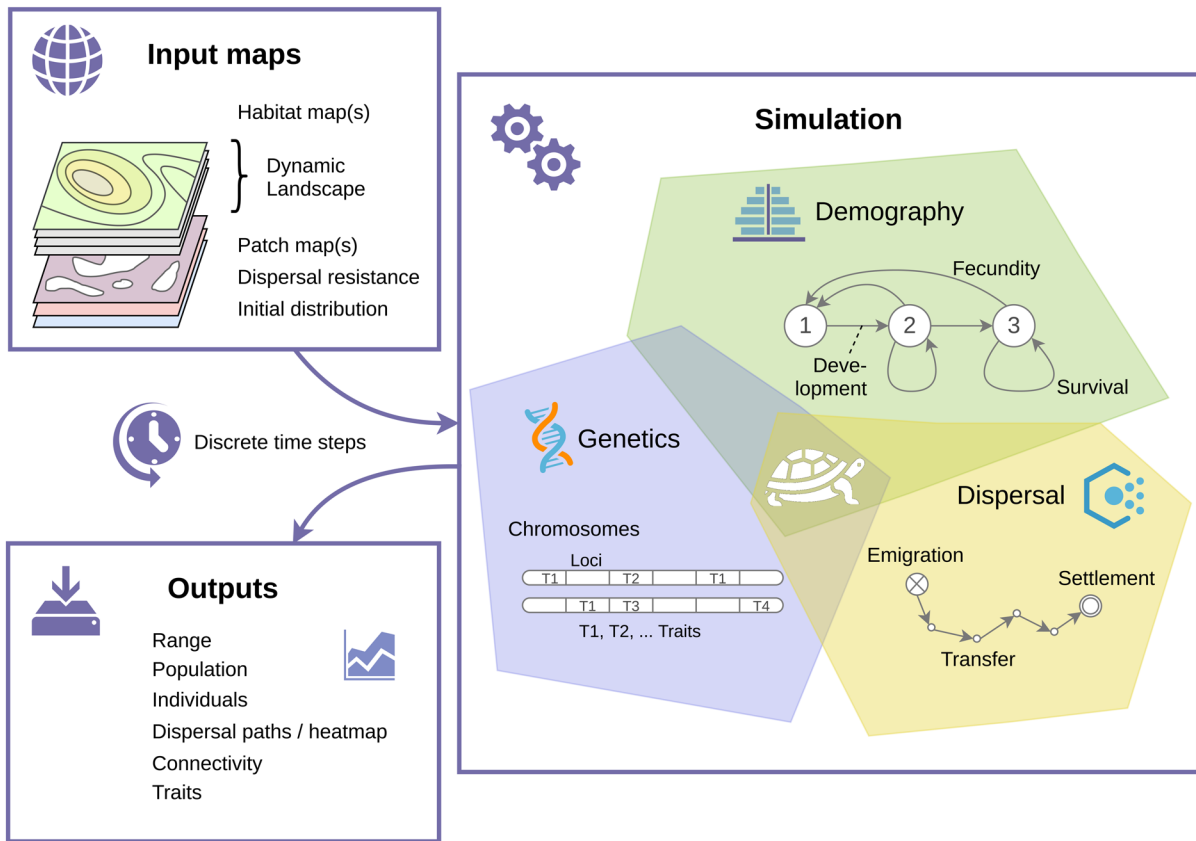
Figure 1. Conceptual overview of a RangeShiftR simulation. The user provides input maps to characterise the landscape, and specifies parameter options that define the three interacting processes of demography, dispersal and evolution (genetics). One option for representing each process is symbolised here as an exemplary model configuration. For example, demography is represented using a stage-structured model with three stages. Different outputs are generated during the simulation and stored in files.

SMS() (stochastic movement simulator; Palmer et al. 2011) or *CorrRW()* (correlated random walk). The choice of the sub-modules thus determines the structure of the species model. The parameters of each model function set the corresponding model parameters, e.g. the maximum growth rate of the population, 'Rmax', in *Demography()*. Apart from the species model, there is a module that handles the input of the landscape, providing two alternative model functions for importing a raster map, *ImportedLandscape()*, or generating an artificial landscape internally, *ArtificialLandscape()*. Two more modules determine the *Initialisation()* of the simulation as well as some general *Simulation()* settings.

The choices made when selecting certain (sub-)modules and specifying their parameters collectively define a RangeShiftR simulation. However, there exists a number of interdependencies among the modules as well as certain compatibility restrictions with some options (Bocedi et al. 2014). To cover them, there is a *ParameterMaster* class whose constructor *RSsim()* takes and consolidates all components of the model and gives informative error messages or warnings to the user in case of incompatible parameter settings. An object of this class defines a RangeShiftR simulation uniquely and can optionally contain a set seed for the random number generator. Using *RunRS()* on the *ParameterMaster* runs the simulation. The

set of model functions constitutes the R interface to the C++ core code, which offers the functionalities of the RangeShifter platform for use from within R while ensuring high computational performance. To integrate the C++ code, the package uses Rcpp ($\geq$ 1.0.0) (Eddelbuettel et al. 2011).

The run time and memory requirements of a RangeShifter simulation can vary widely. Both depend on the number of modelled individuals as well as the represented detail. For example, simulating a movement process involves many more steps than using a dispersal kernel for the transfer phase, and including the genetics module means that the genome of each individual has to be stored. The writing of output files contributes significantly to the run time, and it is recommended to generate only necessary output.

Helper functions

To aid parameter specification, RangeShiftR includes additional helper functions to estimate or visualise the effect of some parameters (Fig. 2). The function *plotProbs()* can be used on a demography or dispersal (sub-)module to plot the shape of density-dependent relationships, for example the fecundity or emigration probability. Most modules in a RangeShiftR simulation influence each other either directly
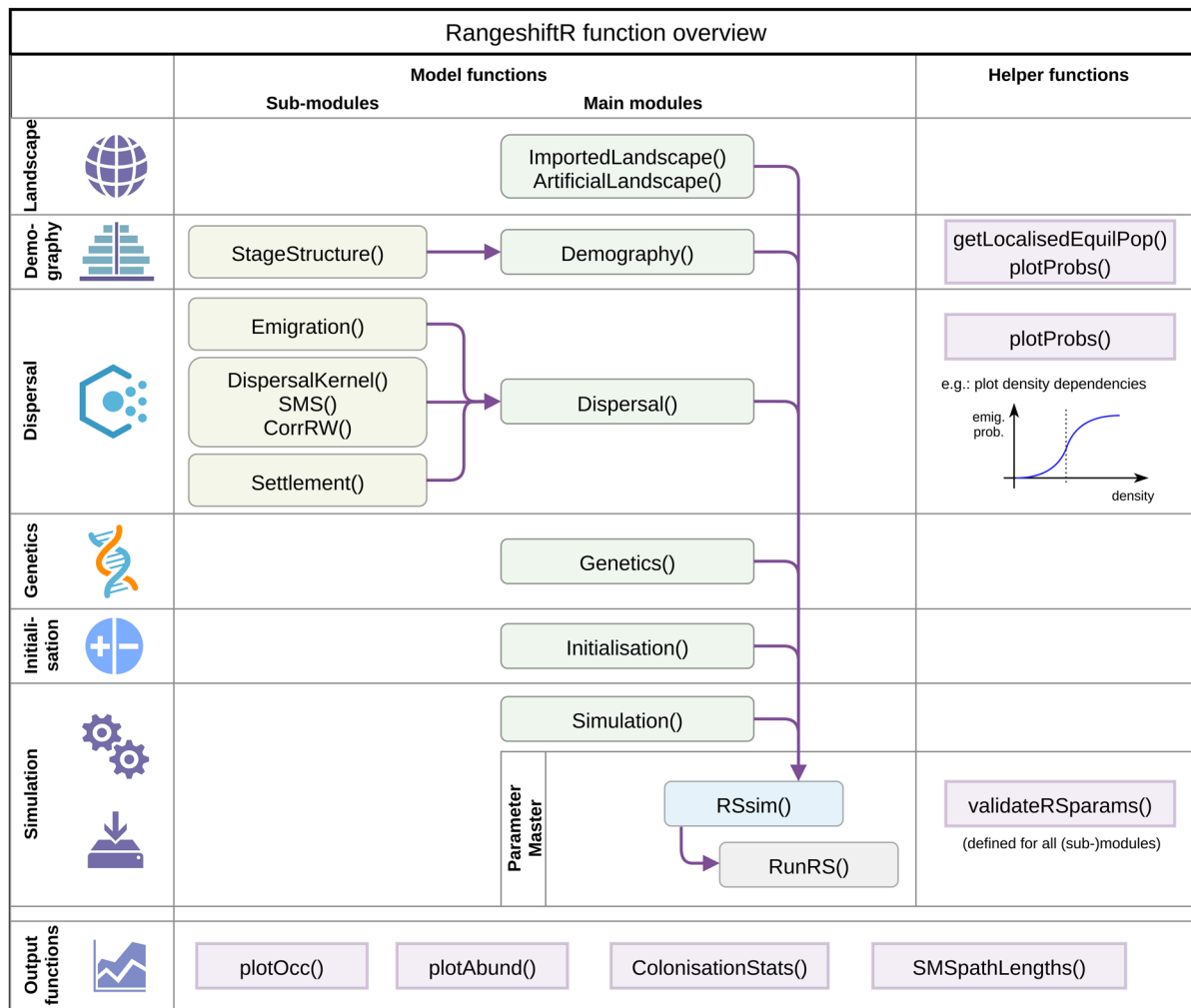
Figure 2. RangeShiftR function overview. The first column introduces the various modules with their respective icons, as reference to Fig. 1. The rounded boxes and arrows in columns 2 and 3 indicate model functions and their respective hierarchical relations. They are class constructors used to define the sub-modules (column 2, yellow) and main modules (column 3, green), which can be combined to a parameter master (blue) to compose the RangeShiftR model. The function RunRS (grey) then starts the simulation. The angled boxes in the last column indicate helper functions that are related to their respective modules. The angled boxes in the bottom row are separate from the columns and itemise the output functions that can be used for processing the simulation results.

or indirectly, and certain parameters may have implications in various places. Therefore, it can prove challenging to express knowledge about the system by directly specifying separate numerical parameter values. For example, specifying a stage-structured demographic model requires estimates for at least one parameter that determines the nature of density-dependence in survival or fecundity. We typically will not have a direct estimate for that parameter, but are much more likely to have an estimate of overall carrying capacity or equilibrium density, which in the model is an emergent outcome of all the demographic parameters. Thus, to guide the choice of suitable parameter values, RangeShiftR contains the novel function *getLocalisedEquilPop()* to estimate the combined effect of density-dependent population dynamics on a closed population (cf. example below). Finally, the function *validateRSparams()* can be used on any (sub-)module to check if all parameters are set within their admissible ranges.

Output functions

All simulation output is written to text files in the formats provided by the RangeShifter platform. The RangeShiftR package also includes dedicated output functions that facilitate the inspection of these results by processing and visualising the output files. These include *plotOcc()* and *plotAbund()* to show the simulated time series of occupancy and abundance, *ColonisationsStats()* for the computation of spatial statistics such as the occupancy probability and the time to colonisation, and *SMSpathLengths()* to display the distribution of dispersal path lengths. A novel output option is provided by the creation of dispersal heatmaps for SMS in the form of raster files, which show the number of dispersers that passed through each location and can be readily processed and plotted with R. Some output functions use basic functionality from the raster package (≥ 3.0.0) (Hijmans and van

Etten 2016) to generate and plot maps. All documentation pages use Rdpack (≥ 0.7) to include references.

Simulation modules

In the modular structure of RangeShiftR, each module represents a different aspect of the simulation (Fig. 2), allowing for adaptable levels of model complexity. Below, the main modules are described briefly. For comprehensive documentation, covering all parameters and options, we refer to the package documentation and the RangeShifter manual (Bocedi et al. 2014, 2021; <https://rangeshifter.github.io>).

Landscape

A RangeShiftR simulation runs on a cartesian grid in which each cell holds information about its cover. This is described by a single layer that represents cells of either a land class or a habitat quality score ranging from zero to 100%. To relate this layer to a cell's habitat suitability for the study species, a value for its demographic density-dependence must be provided: either one for each land class or that for a 100% habitat cell.

There are two variants of the landscape module: Usually, the landscape map will be imported from an ASCII raster file but it can also be artificially created by a built-in function. Imported landscapes have additional options: they can be patch-based, in which case a second raster file is required to indicate each cell's patch ID. Additionally, a raster of dispersal resistance values and a presence–absence raster of the initial distribution can be loaded. With the new functionality of dynamic landscapes introduced in RangeShifter 2.0, the cover, patch and dispersal resistance layer can be changed at any given year during the simulation.

Demography

The modelled demography is determined by two main choices: Firstly, the population can have overlapping or non-overlapping generations, meaning it can be stage-structured or not. In the former case, the sub-processes fecundity, survival and development are explicitly simulated each year, whereas in the latter case only fecundity is modelled. Depending on this choice, the value of the demographic density-dependence defined in the landscape module is interpreted differently: for a stage-structured population it represents the strength of demographic density-dependence (1/b), which can act on all three sub-processes, while for a non-structured population it is interpreted as the carrying capacity (K). A stage-structure is an optional sub-module that is represented by its own class and that can be added to the demography module and which allows various parameters in the demography and dispersal modules to be stage-specific. Secondly, the population can be modelled as sexual or asexual. In sexual models, individuals are characterised by their sex so that various parameters can be sex-specific and the reproductive dynamics may include an explicit mating system. Asexual models can be applied to asexually reproducing species or to species for which only the female sex is modelled as they are assumed to be the limiting sex for the demographic or spatial dynamics.

Dispersal

The dispersal module has three obligatory sub-modules, which represent the explicitly modelled phases of dispersal (Travis et al. 2012). The first phase is emigration, in which an individual decides whether to leave its natal cell or patch. During the subsequent transfer phase the individual moves through the landscape, which can be simulated by three alternative methods: either with a dispersal kernel or with explicit consideration of the movement processes using the stochastic movement simulator (SMS; Palmer et al. 2011) or a correlated random walk. The dispersal event is concluded with the settlement phase, when the individual stops in a different habitable cell or patch. Individuals are allowed to disperse only once during their lives. Various dispersal attributes can be, if applicable, sex- and stage-specific, and the emigration and settlement probabilities can additionally be density-dependent (using the value for demographic density-dependence given in the landscape module). The modelling options for the settlement decision depend on the chosen transfer method and may include an option of mate finding.

Genetics

Individuals can carry a genome that they inherit from their parent(s) at birth. The genome may consist of multiple autosomal loci that can either be neutral or coding for traits (Bocedi et al. 2021). Currently, only the dispersal parameters can be treated as heritable traits, which allows evolution of dispersal strategies. The genetic architecture is highly flexible and processes such as recombination, mutation and pleiotropy can be explicitly modelled. Modelling of neutral loci allows explicit and individual-based population genetic simulations to address questions on how environmental features and processes, in interaction with population dynamics and dispersal behaviours, shape the genetic structure and diversity of populations (Manel et al. 2003).

Initialisation

The initial state of the simulation in the starting year can be defined in three different ways: with an initial distribution map specified in the landscape module, with a list of individuals and their location, or at a given population density in randomly selected locations.

Simulation

This module specifies the general simulation settings like the number of simulated time steps (years) and replicates, the types of generated output and some more specialised options, such as imposing a (shifting) gradient or enabling environmental stochasticity.

Using RangeShiftR

The RangeShiftR 1.0 package can be readily installed from the github repository 'RangeShifter/RangeShiftR-package'. As a widely applicable simulation software, RangeShiftR aims to provide easy access via a range of resources to support the user: all functions are comprehensively documented on R help pages, an extensive user manual is available online, and the webpage (<https://rangeshifter.github.io>) features a support forum as well as a collection of detailed tutorials that illustrate the model's scope and introduce the available modelling options. The tutorials include adaptations of the three original RangeShifter examples (Bocedi et al. 2014), accompanied by sample code for analysis and visualisation. Additionally, we provide a fourth tutorial that demonstrates novel features of RangeShifter 2.0 (Bocedi et al. 2021) by simulating the range dynamics of a species in a changing landscape. Here, we present a shortened form of this fourth tutorial as an example to introduce the RangeShiftR syntax. All required input files can be found on our webpage or downloaded directly via <https://rangeshifter.github.io/RangeshiftR-tutorials/files/Tutorial3_Inputs.zip>.

Landscape

When using the novel RangeShifter feature of dynamic landscapes, we specify the file names of the changing habitat maps, their corresponding patch files, and the order of years in which these become effective. All maps are imported as ASCII rasters by the function *ImportedLandscape()*. Further arguments are the (optional) map of initial distribution, the number of land cover types 'Nhabitats', as well as their respective demographic density-dependence 'K_or_DensDep'.

```
landnames<-c("map_01.asc","map_02.asc","map_03.
asc","map_04.asc")
pchs <- c("patches_01.asc","patches_02.asc",
"patches_03.asc","patches_03.asc")

land <- ImportedLandscape
    (LandscapeFile=landnames,
    PatchFile=pchs,
    DynamicLandYears=c(0,80,110,140),
    SpDistFile="init_dist.asc",
    Nhabitats=5,
    Resolution=10,
    K_or_DensDep=c(125,0,150,75,0),
    SpDistResolution=10)
```

Demography

The population model is set up to use explicit sexes and a stage-structure, i.e. generations are overlapping. In the *Demography()* module the coded argument 'ReproductionType' determines whether both sexes are modelled. The *StageStructure()* sub-module takes the transition matrix and can set optional density-dependencies on the sub-processes of fecundity, survival and development.

```
TraMa <- matrix(c(0.0, 0.0, 0.0, 5.0,
    1.0, 0.1, 0.0, 0.0,
    0.0, 0.6, 0.2, 0.0,
    0.0, 0.0, 0.45, 0.85),
    ncol=4, byrow=TRUE)
demog <- Demography(ReproductionType=1,
    StageStruct=StageStructure(Stages=4,
        TransMatrix=TraMa,
        FecDensDep=T,
        SurvDensDep=T,
        SurvDensCoeff=0.4))
```

The helper function *getLocalisedEquilPop()* can assist in understanding how the demographic rates set in the demography module and the local density-dependence (1/b) affect the simulated abundances:

```
getLocalisedEquilPop(demog=demog,
DensDep_values=seq(50,300,50))
```

It simulates a time series of the population density (in individuals per hectare) of a single closed population for varying values of 1/b (given by 'DensDep_values'). This is achieved by repeated matrix multiplication with the density-dependent transition matrix until an equilibrium is reached. The function returns these equilibrium densities by stages at the given density-dependence values and generates a bar graph (Fig. 3a). The generated densities approximate the equilibrium densities of a closed patch in the RangeShiftR simulation, and can thus be used to guide the choice of the parameter 1/b. However, the matrix approach neglects stochasticity, the scheduling of survival and reproduction and the integer units of abundance, so that the quality of the estimate is lower for smaller populations.

Dispersal

The three phases of dispersal are first defined independently as sub-modules before assembling them in the dispersal module. In the *Emigration()* sub-module, the emigration probability is modelled as stage- and density-dependent, therefore we provide a matrix with one row per stage containing three parameters each, which define how emigration probability relates to population density:

```
emig <- Emigration(StageDep=T,
    DensDep=T,
    EmigProb=cbind(0:3, c(0.55,0.45,0,0),
        c(5,5,0,0), c(1,1,0,0)))
```

The transition phase uses *SMS()* as a method. It implements a step-wise process which is defined by the directional persistence ('DP') and size of memory ('MemSize'). Additionally, we set a dispersal bias (second line) that describes the strength and decay of an additional bias to move away from the original patch, as well as habitat-specific dispersal resistances and a constant per-step mortality:

(a) Localised equilibrium densities　　　　　　**(b) Time to colonisation**
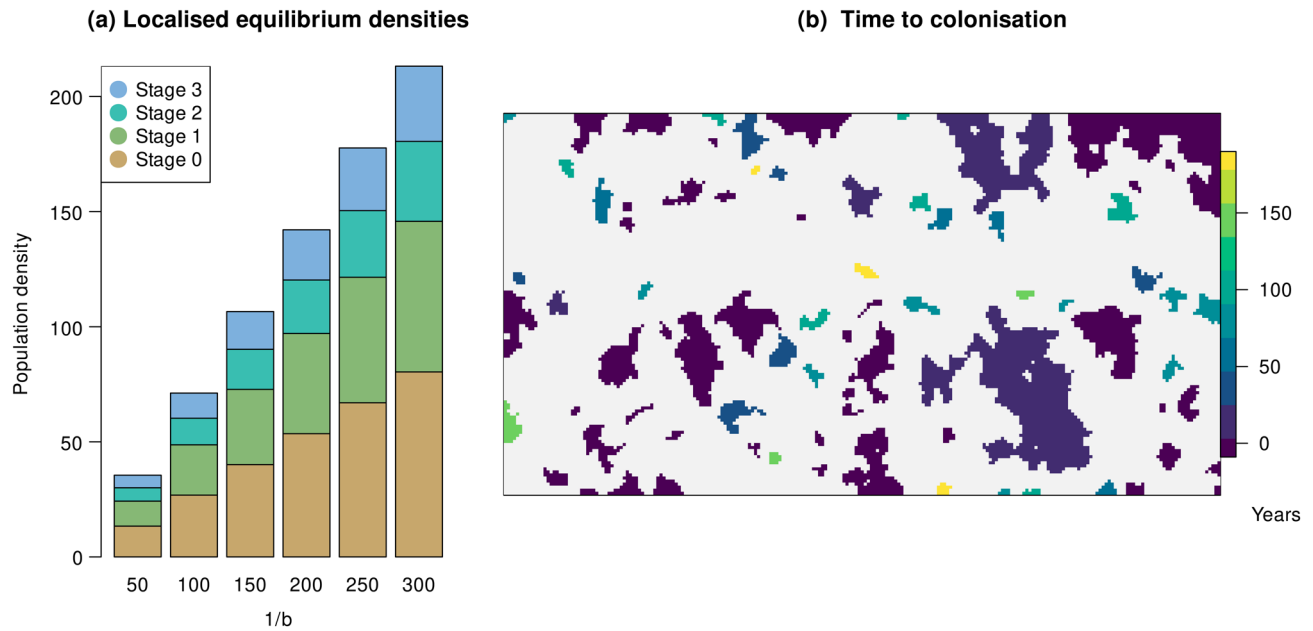

Figure 3. RangeShiftR example. (a) Bar graph generated by the helper function *getLocalisedEquilPop()*, showing the localised equilibrium densities classified by stages over the parameter 1/b (both in units of Inds/ha). They serve as a quick approximation to assess the effect of density-dependent demographic rates. (b) Raster generated by the output function *ColonisationStats()*, showing the average time to colonisation.

```
tran <- SMS(DP=1.8, MemSize=4,
    GoalType=2,   GoalBias=2.5,   AlphaDB=.4,
    BetaDB=10,
    Costs=c(3,5,1,2,30),
    StepMort=0.01)
```

The *Settlement()* sub-module defines the minimum and maximum number of steps permitted and sets the mate-finding requirement:

```
sett <- Settlement(MaxSteps=80, MinSteps=15,
    MaxStepsYear=20,
    FindMate=T)
```

Now, the previously defined sub-modules can be combined in the *Dispersal()* module:

```
disp <- Dispersal(Emigration=emig,
    Transfer=tran,
    Settlement=sett)
```

Genetics

The *Genetics()* module is optional and we leave it disabled here (but see Bocedi et al. 2021 and the online tutorials for an example of this functionality). Although this implies missing inter-individual variation in dispersal traits in our example, individuals are still characterised by their sex, stage and age, which can affect some demographic and dispersal attributes.

Initialisation

The function *Initialisation()* uses mostly coded arguments to define the spatial distribution and density of the initial population. The simulation in this example is initialised in all locations indicated by the initial distribution map (provided to the landscape module) at the density given in 'IndsHaCell'. Further, the stage- and age-distributions of the initial population are set.

```
init <- Initialise(InitType=1,
    SpType=0,
    InitDens=2,
    IndsHaCell=75,
    PropStages=c(0,0.6,0.2,0.2),
    InitAge=2)
```

Simulation

The *Simulation()* runs for 200 years and over twenty replicates. The population, range and SMS paths outputs are enabled and will be generated at the given time intervals:

```
simul <- Simulation(Years=200,
    Replicates=20,
    OutIntPop=5,
    OutIntRange=5)
```

Model run and results

All defined model components are combined into the parameter master with *RSsim()*, which optionally takes a seed to

pass to the random number generator and make the simulation reproducible. Every RangeShiftR simulation is defined by an instance of this class and the path to its directory and is run using *RunRS()*:

```
s <- RSsim(land=land, demog=demog, dispersal=
    disp, init=init,simul=simul,seed=123456)
dirpath <- "RS_example/"
RunRS(s, dirpath=dirpath)
```

The simulation output is written to text files in the 'Outputs' folder of the directory. These can be further processed and visualised using the auxiliary output functions. For example, Fig. 3b shows a result that is returned by the function *ColonisationStats()*. It calculates the time to colonisation and the occupancy probability at given years and can map the values onto the landscape:

```
col <- ColonisationStats(s, dirpath, maps=T)
raster::plot(col$map_col_time)
```

In the resulting plot, the non-suitable landscape matrix appears grey and all habitat patches are coloured according to their averaged time to colonisation over all replicates. In this example, smaller patches tend to get colonised later than larger ones.

Discussion

RangeShiftR 1.0 provides, for the first time, an open-source individual-based, eco-evolutionary simulation platform in R, which includes a diversity of processes and offers various levels of complexity, especially for the demographic and dispersal processes. It gives access to the established RangeShifter platform (Bocedi et al. 2014, 2021), adds supporting functionality for model specification and analysis in R, and provides comprehensive documentation to guide the user.

RangeShiftR complements the existing toolbox of R packages, as it offers some important features that have not been available so far. Existing R implementations of spatially-explicit population modelling frameworks, such as the recently published package steps (Visintin et al. 2020) or the demoniche package (Nenzén et al. 2012), are population-based. In contrast, RangeShiftR is individual-based and hence allows for an explicit representation of genetics and evolutionary dynamics. The package vortexR (Pacioni and Mayer 2017) implements post-analysis functions for the prominent, spatially-implicit, Vortex model (Lacy 1993) that is also individual-based and commonly applied for population viability analysis (PVA). Here, RangeShiftR provides a useful alternative that allows conducting spatially-explicit PVA under more complex dispersal assumptions.

The RangeShifter GUI (Bocedi et al. 2014, 2021) and the RangeShiftR package constitute two complementary entities, as they represent alternative interfaces to the same software core. The GUI version offers an intuitive handling of the model and visual tracking of simulation outcomes, making it particularly suited for the use by stakeholders or for undergraduate education. The RangeShiftR package, on the other hand, is especially useful for research purposes. It offers transparent, reproducible workflows, as the entire simulation can be scripted in R, along with the visualisation and post-analysis of simulation results. This also facilitates large-scale parameter comparisons, as required in sensitivity and robustness analyses. The use of Rcpp (Eddelbuettel et al. 2011) allows running of the simulation in a C++ module and thereby yields high performance, while the integration in R makes RangeShiftR available for multiple platforms and provides the infrastructure for parallel and cluster computing without having to adapt the C++ back end.

RangeShiftR holds many opportunities for interoperation with other R packages. Firstly, it can be readily integrated with packages for describing the landscape context (e.g. raster; Hijmans and van Etten 2016) or species distribution modelling (e.g. biomod2; Thuiller et al. 2009, sdm; Naimi and Araújo 2016). Secondly, it permits coupling of different model types, as exemplified by coupling RangeShifter with the land-use model CRAFTY (Murray-Rust et al. 2014, Synes et al. 2019). Thirdly, it enables integrated use with existing methodological devices, like inverse parameterisation through Bayesian inference, for example using the package BayesianTools (Hartig et al. 2017).

RangeShiftR can help overcome some of the challenges that have prevented more widespread use of mechanistic range models (Briscoe et al. 2019) by offering high accessibility. In the future, we plan to enhance the platform further to improve forecasts under global change. For example, the model currently operates on a single habitat layer that contains either land classes or habitat quality. Therefore, demographic rates are related to the environment only indirectly via the user-defined carrying capacities or density-dependence coefficients. Further, all density-dependent relationships have a predetermined shape that is controlled by specified parameters but cannot be replaced by a user-specified function. Moreover, RangeShiftR currently models only a single species and does not incorporate species interactions. Lastly, the genetics module is currently restricted to modelling evolution of dispersal traits while demographic traits cannot evolve. Thus, potential future extensions of the platform will involve explicitly modelling demography–environment relationships (Pagel and Schurr 2012) species interactions, and genetic evolution of demographic traits. As the code is open source, there is now an opportunity for a broad community of researchers and modellers to contribute to representing these important processes in future versions of the platform.

The RangeShiftR package constitutes an important step towards making frameworks for modelling range dynamics under global change accessible to a wider audience (Schurr et al. 2012, Lurgi et al. 2015, Zurell et al. 2016). We hope that this will inspire a more widespread use of mechanistic distribution models, for example to guide conservation

efforts and ecosystem management, and facilitate more seamless integration with other modelling tools.

To cite RangeShiftR or acknowledge its use, cite this Software note as follows, substituting the version of the application that you used for 'version 1.0':

Malchow, A.-K. et al. 2021. RangeShiftR: an R package for individual-based simulation of spatial eco-evolutionary dynamics and species' responses to environmental changes. – Ecography 44: 1443–1452 (ver. 1.0).

Author contributions

Anne-Kathleen Malchow: Conceptualization (equal); Software (lead); Validation (lead); Visualization (lead); Writing – original draft (lead); Writing – review and editing (lead). **Greta Bocedi**: Conceptualization (supporting); Software (equal); Validation (supporting); Writing – review and editing (equal). **Steve C. F. Palmer**: Conceptualization (supporting); Software (equal); Validation (supporting); Writing – review and editing (equal). **Justin M. J. Travis**: Conceptualization (supporting); Project administration (supporting); Supervision (supporting); Writing – review and editing (equal). **Damaris Zurell**: Conceptualization (equal); Funding acquisition (lead); Software (supporting); Supervision (lead); Validation (supporting); Visualization (supporting); Writing – original draft (supporting); Writing – review and editing (equal).

Data availability statement

Data are available from GitHub: <https://github.com/RangeShifter/RangeShiftR-package> (Malchow et al. 2021).

References

Aben, J. et al. 2016. The importance of realistic dispersal models in conservation planning: application of a novel modelling platform to evaluate management scenarios in an Afrotropical biodiversity hotspot. – J. Appl. Ecol. 53: 1055–1065.

Austin, M. 2007. Species distribution models and ecological theory: a critical assessment and some possible new approaches. – Ecol. Model. 200: 1–19.

Bocedi, G. et al. 2014. RangeShifter: a platform for modelling spatial eco-evolutionary dynamics and species' responses to environmental changes. – Methods Ecol. Evol. 5: 388–396.

Bocedi, G. et al. 2021. RangeShifter 2.0: an extended and enhanced platform for modelling spatial eco-evolutionary dynamics and species' responses to environmental changes. – Ecography 44: 1453–1462.

Briscoe, N. J. et al. 2019. Forecasting species range dynamics with process-explicit models: matching methods to applications. – Ecol. Lett. 22: 1940–1956.

Brondizio, E. et al. 2019. Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services. – IPBES Secretariat, Bonn.

Cabral, J. S. et al. 2017. Mechanistic simulation models in macroecology and biogeography: state-of-art and prospects. – Ecography 40: 267–280.

Connolly, S. R. et al. 2017. Process, mechanism and modeling in macroecology. – Trends Ecol. Evol. 32: 835–844.

Dormann, C. F. et al. 2012. Correlation and process in species distribution models: bridging a dichotomy. – J. Biogeogr. 39: 2119–2131.

Eddelbuettel, D. et al. 2011. Rcpp: seamless R and C++ integration. – J. Stat. Softw. 40: 1–18.

Elith, J. and Leathwick, J. R. 2009. Species distribution models: ecological explanation and prediction across space and time. – Annu. Rev. Ecol. Evol. Syst. 40: 677–697.

Elith, J. et al. 2010. The art of modelling range-shifting species. – Methods Ecol. Evol. 1: 330–342.

Guisan, A. and Thuiller, W. 2005. Predicting species distribution: offering more than simple habitat models. – Ecol. Lett. 8: 993–1009.

Guisan, A. and Zimmermann, N. E. 2000. Predictive habitat distribution models in ecology. – Ecol. Model. 135: 147–186.

Hartig, F. et al. 2017. BayesianTools: general-purpose MCMC and SMC samplers and tools for Bayesian statistics. – R package ver 0.1-7, <https://CRAN.R-project.org/package=BayesianTools>.

Henry, R. C. et al. 2017. Tree loss impacts on ecological connectivity: developing models for assessment. – Ecol. Inform. 42: 90–99.

Hijmans, R. J. and van Etten, J. 2016. raster: geographic data analysis and modeling. – R package ver 3.4-5, <https://CRAN.R-project.org/package=raster>.

Hijmans, R. J. et al. 2017. dismo: species distribution modeling. – R package ver 1.3-3, <https://CRAN.R-project.org/package=dismo>.

Jarnevich, C. S. et al. 2015. Caveats for correlative species distribution modeling. – Ecol. Inform. 29: 6–15.

Kearney, M. and Porter, W. 2009. Mechanistic niche modelling: combining physiological and spatial data to predict species' ranges. – Ecol. Lett. 12: 334–350.

Lacy, R. C. 1993. VORTEX: a computer simulation model for population viability analysis. – Wildl. Res. 20: 45–65.

Lurgi, M. et al. 2015. Modelling range dynamics under global change: which framework and why? – Methods Ecol. Evol. 6: 247–256.

Manel, S. et al. 2003. Landscape genetics: combining landscape ecology and population genetics. – Trends Ecol. Evol. 18: 189–197.

Malchow, A.-K. et al. 2021. Data from: RangeShiftR: an R package for individual-based simulation of spatial eco-evolutionary dynamics and species' responses to environmental changes. – GitHub, <https://github.com/RangeShifter/RangeShiftR-package>.

Martínez-Minaya, J. et al. 2018. Species distribution modeling: a statistical review with focus in spatio-temporal issues. – Stoch. Environ. Res. Risk Assess. 32: 3227–3244.

McGill, B. J. et al. 2015. Fifteen forms of biodiversity trend in the Anthropocene. – Trends Ecol. Evol. 30: 104–113.

Murray-Rust, D. et al. 2014. Combining agent functional types, capitals and services to model land use dynamics. – Environ. Model. Softw. 59: 187–201.

Naimi, B. and Araújo, M. B. 2016. sdm: a reproducible and extensible R platform for species distribution modelling. – Ecography 39: 368–375.

Nenzén, H. K. et al. 2012. demoniche – an R-package for simulating spatially-explicit population dynamics. – Ecography 35: 577–580.

Pacioni, C. and Mayer, F. 2017. vortexR: an R package for post Vortex simulation analysis. – Methods Ecol. Evol. 8: 1477–1481.

Pagel, J. and Schurr, F. M. 2012. Forecasting species ranges by statistical estimation of ecological niches and spatial population dynamics. – Global Ecol. Biogeogr. 21: 293–304.

Palmer, S. C. F. et al. 2011. Introducing a 'stochastic movement simulator' for estimating habitat connectivity. – Methods Ecol. Evol. 2: 258–268.

Phillips, S. J. et al. 2017. Opening the black box: an open-source release of Maxent. – Ecography 40: 887–893.

Qiao, H. et al. 2015. No silver bullets in correlative ecological niche modelling: insights from testing among many potential algorithms for niche estimation. – Methods Ecol. Evol. 6: 1126–1136.

Schurr, F. M. et al. 2012. How to understand species' niches and range dynamics: a demographic research agenda for biogeography. – J. Biogeogr. 39: 2146–2162.

Synes, N. W. et al. 2019. Coupled land use and ecological models reveal emergence and feedbacks in socio-ecological systems. – Ecography 42: 814–825.

Thuiller, W. et al. 2009. BIOMOD – a platform for ensemble forecasting of species distributions. – Ecography 32: 369–373.

Travis, J. M. J. et al. 2012. Modelling dispersal: an eco-evolutionary framework incorporating emigration, movement, settlement behaviour and the multiple costs involved. – Methods Ecol. Evol. 3: 628–641.

Urban, M. C. et al. 2016. Improving the forecast for biodiversity under climate change. – Science 353: 6304.

Visintin, C. et al. 2020. steps: software for spatially and temporally explicit population simulations. – Methods Ecol. Evol. 11: 596–603.

Zurell, D. et al. 2016. Benchmarking novel approaches for modelling species range dynamics. – Global Change Biol. 22: 2651–2664.