Objects, clones and collections: ecological models and scenario analysis with simecol

Thomas Petzoldt[a]

R is increasingly accepted as one of the standard environments for ecological data analysis and ecological modeling. An inreasing collection of packages explicitly developed for ecological applications (Kneib and Petzoldt, 2007) and a number of textbooks that use R to teach ecological modeling (Ellner and Guckenheimer, 2006; Bolker, 2007) are just an indicator for this trend. In this context, the package **simecol** (simulation of ecological models) was developed in order to facilitate implementation, analysis and share of simulation models by means of object oriented programming (OOP) with S4 classes.

The idea behind **simecol** is an object model of ecological models, i.e. to put everything needed (state variables, parameters, inputs, equations) to define an ecological model together in one code object (an instance of a subclass of `simObj`), that can be handled by appropriate generic functions.

Because all essentials of a particular model are encapsulated in one code object, individual instances can simply be copied with the assignment operator `<-`. These clones can be modified with accessing functions to derive variants and scenarios without copying and pasting source code. This way, it is also possible to interactively enable or disable online-visualisation (using observer-slots), to adapt numerical accuracy or to compare scenarios with different structure, e.g. ecological models with different types of functional response.

After a short overview the presentation will concentrate on examples how to clone, modify and extend **simecol** objects and how to organize scenario analyses. From the user's perspective these are:

1. Implement a model-prototype by filling out a pre-defined structure or by modifying existing examples,
2. Simulate and test your model with existing solvers or develop your own algorithms,
3. Clone your prototype object and modify data and/or code of individual clones to generate scenarios,
4. Simulate, analyse, compare scenarios, fit parameters, supply observer functions for run-time visualisation.
5. Save your model object and share it with your colleagues, students and readers of your papers.

Individual **simecol**-objects can be stored persistently as binaries or human-readable list representation which can be distributed in reproducible and fully functional form. In addition, it is also possible to assemble collections of models as separate R-packages, together with necessary documentation and examples, e.g. to reproduce the figures of a paper, or with additional classes and functions extending **simecol**.

References

Bolker, B. (2007). *Ecological Models and Data in R.* Princeton University Press, Princeton. in press.

Ellner, S. P. and Guckenheimer, J. (2006). *Dynamic Models in Biology.* Princeton University Press.

Kneib, T. and Petzoldt, T. (2007). Introduction to the special volume on ecology and ecological modeling in R. *Journal of Statistical Software*, 22(1):1–7.

[a] Technische Universität Dresden, Institute of Hydrobiology, 01062 Dresden, Germany, thomas.petzoldt@tu-dresden.de, `http://tu-dresden.de/Members/thomas.petzoldt`