The RTM package

a Learning Environment for Reaction Transport Modelling

Karline Soetaert, Lubos Polerecky and Dries Bonte

June 2021

Reaction Transport Modelling

Since many years (starting in 1998!) I (Karline Soetaert) have been teaching environmental modelling to master students at Ghent University, and since 2018 also at Utrecht University.

The environmental models we make in class are mechanistic models that start from principles of physics, chemistry, biology, ecology, and that are mathematically composed of differential equations. They are so-called "engineering type of models", aiming to advance our understanding of natural systems and that allow prediction of these systems under various scenarios.

The aim of the course is to prepare students so that they are able to make models from scratch if needed. This differs from many modelling courses where students merely learn how to apply existing tools.

More specifically, in this course the students learn to:

- translate environmental problems into mechanistic models that consist of differential equations.
- implement these mathematical equations into the R programming language (R core team, 2020) and solve them.
- interpret the results in terms of the environmental problem.
- apply the models to realistic scenarios.

Students that follow this course are submerged in the fields of physics, chemistry, biology, but also need to be knowledgeable in mathematics and computer science.

We use R (R core team 2020) as the language for implementing and solving these models, mainly because the R-packages we use offer an "ecosystem" of methods to work with differential equations. They are the packages rootSolve, deSolve, ReacTran (Soetaert, 2009; Soetaert et al., 2010; Soetaert and Meysman, 2012). The mathematical methods in these packages have been written so that they hide much of the details, allowing students to focus on the original problem rather than being concerned with how to tweak or even implement the numerical solutions. Accessory functions facilitate plotting solutions, calculating summaries, fitting models to data, etc.

Our students typically follow a master in oceanography, biology, biogeochemistry, hydrology or physics. Some have a good understanding of the processes occurring in the natural environment, but have never written a computer program before; others have little process knowledge but ample experience in programming.

Due to this variation in skills, needless to say that it is difficult to find a teaching style that is suitable for all of the students.

Our way of teaching: flipped classroom

In the *traditional learning environment*, the instructor explains the context, theory and examples during lectures, and encourages students to practice the taught material by themselves at home.

In a *flipped classroom*, this process is reversed. Students obtain the theoretical lectures in advance and prepare for class first. Then, while in the classroom, they put the learned concepts into practice by solving problems and exercises with a direct feedback from the tutor.

One of the advantages of a flipped classroom is that students can work more focused and obtain a more thorough understanding of the material. Typically the level of understanding of students in the modelling class is highly diverse. Thus, in the traditional way of teaching, the brighter students often get bored as the material is being taught too slowly, while less bright students become demotivated or frustrated as they cannot follow. In the flipped classroom concept, students can take the time they need to prepare for lessons. Subsequently, in the classroom, they directly interact with tutors and can therefore explore the topics in greater depths. This provides more room for tutoring "a la tete du client".

The modelling course taught by Karline Soetaert and Dries Bonte at Ghent University has been in the form of a flipped classroom since the academic year 2019-2020. This kind of teaching proved to be a positive experience: whereas in previous years students would complain that they do not get enough practice in class, there is now ample time for practical exercises. Also the class is much more dynamic, with dramatically improved students participation and involvement.

In 2021, Lubos Polerecky and Karline Soetaert also implemented the flipped classroom concept in the RTM course given at Utrecht University.

Implementation of the flipped classroom

To prepare for each of the lessons, the students are offered the choice to either read one or more chapters from a book (Soetaert and Herman, 2009), or watch a video that explains the theory. Then the knowledge of the theory is tested by means of a number of short (often multiple-choice) questions.

In class then, more substantial exercises are given under supervision of the tutors.

Video's that elaborate on the theory

One way of "flipping" the classroom is by means of video lessons that elaborate on the theory. There is a series of videos for all the material taught in the modelling class.

The videos typically take around 15 minutes or less, and are created as follows:

- (1) first didactically sound slide presentations are created,
- (2) the narrative of the material on the slides is written down, and
- (3) the spoken text is recorded while playing the slides.

Steps (1) and (2) need substantial input from the tutors. Step (3) can be outsourced.

Apart from the direct benefits, this teaching approach also offers a number of longer-term advantages. The video material has been put on the internet (i.c., youtube) to serve as an online course.

Also, it can be used by students at any time to refresh their knowledge on a subject. Students can move through these videos at their own pace - some complain that they go too slow - multi-tasking or the fast-forward knob is then an option.

Quizzes and questions

Each of the video lessons is accompanied by a set of quizzes (questions), and small exercises, so that the student can test her/his knowledge on the subject.

Before we introduced thes quizzes, we often noted that some students overestimate their grasp of a subject, and are overly confident. Other students are baffled by the complexity of the material and may loose confidence that they eventually will manage the topic. As one student said (in his/her evaluation): "I loved this course, but the pressure was immense".

The questions and small exercises help them assess how far they have understood the lessons, and also prepares them more thoroughly for the exercises in class, and -ultimately- for the exam. Another aim is to improve students' quantitative problem-solving skills, by giving them hints and guidance as to how they can tackle a certain question.

The implementation of the questions has been done in the form of 'tutorials' as provided from the *learnr* package (Schloerke et al., 2020). They consist of multiple-choice questions, with here and there a more substantial exercise that the students have to solve using R code.

There is no grading of these tutorials, so the students may use them for their own benefit. However, as they also serve to prepare for some of the exam questions, this hopefully gives extra impetus to actually conduct the quizzes and make the exercises.

Exercises

During class the students have to work in small groups on practical modelling applications. The models deal with a variety of environmental issues. Care has also been taken to make the approaches in the exercises versatile.

At the time of writing this vignette, the exercises dealt, a.o. with the following environmental issues:

- *The CO2 problem.* The atmospheric CO2 concentrations are described with an *earth-system box model* of carbon fluxes amongst the large earth compartments. The impact of anthropogenic activities on the atmospheric CO2 concentration is estimated. Effects of some mitigation strategies is calculated.
- *The ozone problem.* The ozone concentration in the lower atmosphere is modeled. Ozone is a green-house gas and at high concentrations ozone is harmful to humans and animals. It also plays an important role in absorbing UV radiation. The model is a good example of a typical *chemistry model*, and is used to test how anthropogenic emissions of NO (nitric oxide) due to combustion or burning of fuels alter the natural ozone dynamics.
- *Ocean acidification.* This is a very complex problem as it deals with *equilibrium chemistry.* The topic is introduced by three exercises of increasing complexity. A more extensive model of ocean acidification is not part of the mandatory material, but is dealt with in the accessory material (see below) that students may consult if they are interested in the topic.

- *The COVID pandemic.* A set of *population, epidemiological* (so-called SIR) models are implemented to describe the spread of the SARS-CoV2 (corona) virus, including the hospitalisations and mortality rates. The students practice how to use information from epidemiologists (that are not modelers) in their models.
- *Agricultural eutrophication.* This model application, that describes how fertilization with nutrients affects the *competition* between crops and weeds, also contains a (rather simplistic) *economic component* that accounts for the profits farming can bring.
- *Low oxygen in a river.* This is a 1-dimensional *reaction transport model* that describes the evolution of the oxygen concentration in a river, that is subjected to excessive nutrient input (ammonia). The model is used to estimate the level in nutrient inputs that keep the minimal oxygen concentration above a certain critical level.

Other material is:

- Introduction to R for modellers
- Making conceptual schemes and creating mass balance equations
- The dissolution kinetics of Silica particles
- Bacterial decay of detritus
- Early diagenesis in marine sediments (the coupled cycles of C, N, O, S).
- Competition between plants in grasslands.
- Competition between floating plants and algae in shallow lakes
- Virus dynamics in the ocean
- Bioeconomic modelling of a Scallop aquaculture

New material may be added in the future, thus creating a repository from which modules can be selected based on students interests that vary from year to year.

Accessory material

This consists of a number of small documents that go into somewhat more detail for various topics. It is not compulsory material, so the students get to see this after the exam, and before they start the projects (see below).

These "readers" are meant to satisfy the curiosity of some students that have a desire to go more into depth for certain topic.

They comprise documents dealing with:

- How to implement *events* in dynamic models developed in R
- How to use *forcing functions* based on data in models developed in R
- How to easily showing *observed data* alongside model results in R
- How to *fit* a 1D reaction-transport model to data in R
- How to *visualise* dynamic outputs from a 1D reaction-transport model in R
- How to include *pH dynamics* in a 1D reaction-transport model in R
- How to mathematically investigate the response of systems to a *perturbation from equilibrium*
- How to make *interactive applications* in R
- The basics of the *numerical methods* used for reaction-transport modelling in R

Longer projects

After having done the written exam, the students work on a specific environmental topic in small groups of 3-4 people. In contrast to the exercises made in class, they do this with less guidance from the teachers. In addition, they also need to provide a report, where they outline the scientific rationale of their work, describe their model, and document the results, much like they would do while writing a scientific paper.

Note that the topics of these projects are NOT part of the RTM package.

About this package

This R-package contains the tutorials, exercises and extra material that the students have to work through during their courses at Utrecht University.

Anyone that want to self-teach modelling can consult the tutorials and make the exercises. The answers are also provided, in a subdirectory under the inst package directory

tutorials

The function *RTMtutorial* runs the tutorials. To list all possible tutorials:

```
RTM:::RTMtutorial("?")
```

```
##                      x                                      description
## 1         introduction                      About the course at Utrecht
## 2                  why                        Why modelling is useful
## 3           conceptual                        Making conceptual models
## 4          massbalance                  Creating mass balance equations
## 5              Rmodels              Dynamic modelling in the R language
## 6           largescale         Large-scale models (e.g. earth's C-cycle)
## 7             chemical                  Elementary chemical reactions
## 8          equilibrium      Equilibrium (reversible) chemical reactions
## 9            enzymatic                             Enzymatic reactions
## 10        partitioning   Chemical reactions partitioning between phases
## 11             ecology                            Ecological reactions
## 12  transportProcesses                    The general transport equation
## 13     transportFluxes               Advection and diffusion/dispersion
## 14     transportPorous               Reaction transport in porous media
## 15 transportBoundaries            Boundary conditions in transport models
## 16          transportR Modelling Reaction tranport in porous media in R
```

At the start of each tutorial is a link to the youtube video that deals with the theory.

Exercises

R-function *RTMexercise* opens the PDF with the questions for each exercise. To list all possible exercises:

```
RTM:::RTMexercise("?")
```

```
##                      x                                              description
## 1            modelersR                               Learning R for modellers
## 2           conceptual             Translating problems into a conceptual scheme
## 3          massbalance                          Creating mass balance equations
## 4  massbalance_ecology               Creating mass balance equations in ecology
## 5          carbonCycle                  An earth-system box model of the C-cycle
## 6                ozone                        Ozone dynamics in the troposphere
## 7        dissolutionSi                     Dissolution kinetics of Si particles
## 8       equilibriumNH3                  Equilibrium chemistry - ammonium/ammonia
## 9      equilibriumHCO3              Equilibrium chemistry - the carbonate system
## 10      equilibriumOMD Equilibrium chemistry - impact of mineralisation on pH
## 11            detritus              Bacterial decay of detritus (biogeochemistry)
## 12               COVID                  The COVID pandemic (population dynamics)
## 13               virus                            Virus dynamics in the ocean
## 14                npzd                       NPZD model (marine ecosystem model)
## 15   plant_coexistence     Competition and coexistence of plants in grasslands
## 16          crops_weed      Crops and weed competition (agriculture-economics)
## 17      hyacinth_algae Competition between floating plants and algae in a lake
## 18         riverAnoxia     Anoxia in an estuary (1-D reaction transport model)
## 19         Pdiagenesis        Simple phosphorus diagenesis in marine sediments
## 20          diagenesis    Complex diagenesis in marine sediments (C, N, O2, S)
```

A number of model templates is available and can be downloaded from the Rstudio menu: "File->new File->Rmarkdown..->from template->name_of_the_template".

Accessory material

R-function *RTMreader* opens the PDF with the accessory (non-mandatory) material. To list all possible documents:

```
RTM::RTMreader("?")
```

```
##                   x
## 1            events
## 2          forcings
## 3      observations
## 4           fitting
## 5     visualisation
## 6        pHprofiles
## 7    perturbation_I
## 8   perturbation_II
## 9       interactive
## 10       numericalR
## 11 git_sharing_code
##                                                   description
## 1                          Events in dynamic models developed in R
## 2          Forcing functions based on data in models developed in R
## 3                Showing observed data alongside model results in R
## 4                Fitting a 1D reaction-transport model to data in R
## 5       Visualising outputs from a 1D reaction-transport model in R
## 6               Estimating pH in a 1D reaction-transport model in R
## 7   Response of systems to a perturbation from equilibrium - Part I
## 8  Response of systems to a perturbation from equilibrium - Part II
## 9                                     Interactive applications in R
## 10     Numerical methods used for reaction-transport modelling in R
## 11     Git, GitLab/Github and RStudio - sharing code with the world
```

References

R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Barret Schloerke, JJ Allaire and Barbara Borges (2020). learnr: Interactive Tutorials for R. R package version 0.10.1. https://CRAN.R-project.org/package=learnr

Soetaert Karline (2009). rootSolve: Nonlinear root finding, equilibrium and steady-state analysis of ordinary differential equations. R-package version 1.6

Soetaert Karline, Thomas Petzoldt, R. Woodrow Setzer (2010). Solving Differential Equations in R: Package deSolve. Journal of Statistical Software, 33(9), 1–25. URL http://www.jstatsoft.org/v33/i09/ DOI 10.18637/jss.v033.i09

Soetaert, Karline and Meysman, Filip (2012). Reactive transport in aquatic ecosystems: Rapid model prototyping in the open source software R Environmental Modelling & Software, 32, 49-60.