Response to reviewers

“Hydrology modelling R packages: a unified analysis of models and practicalities from a user perspective”

Previous responses from the online discussion are in blue. New responses after revision are in red.

Response to Editor

“Editor Decision: Publish subject to revisions (further review by editor and referees) (03 Feb 2021) by Jan Seibert

Comments to the Author: The reviewers provide useful suggestions on how to improve this manuscript. Based on the responses in the discussion phase, it seems to me that the authors will be able to address most of the comments. However, there is the fundamental issue raised by reviewer#1 whether this manuscript fits best in HESS or whether a journal like GMD might be a better fit. Here I am less convinced by the reply of the authors. This response mainly states that one of the chief editors had agreed to consider this a technical note for HESS but does not provide much on why this decision was taken. Besides the argument in the response, I would also like to ask the authors to clarify in the text why this paper fits into HESS as a technical note. While I sympathize with the arguments of the authors, re-reading the info on https://www.hydrology-and-earth-system-sciences.net/about/manuscript_types.html one has to admit that the reviewer has point here.”

We would like to thank the Editor for his positive comments on the manuscript. In the revised version, we have addressed most of the reviewers’ comments. We have especially emphasized the place of R for hydrological modelling and how Hydrology modelling R packages are relevant for the hydrological science community. We have also tried to better justify the choice of HESS for publishing this article. The manuscript now contains a Methodology section and sub-headers to make it much easier to read. Please note that we also made a couple of minor modifications. Namely, we replaced “hydrological response units” with "hydrological similarity units" for the dynamic TOPMODEL (see Beven and Freer, 2001), we added a sentence about the concept of equifinality in Sect. 5.1.1 and a sentence about the commitment of R-Hydro developers in Sect. 6.3. We also corrected some typos and rephrased some sentences.

Response to Reviewer 1

We would like to thank Reviewer 1 for his/her constructive comments on the manuscript. We provide an answer to each comment hereafter.

“1. My main problem with this paper is that it does not appear to fit the scope of HESS: https://www.hydrology-and-earth-system-sciences.net/about/aims_and_scope.html as it is essentially a comparison of modelling packages from an user perspective, rather than a study that introduces new methods or provides any new process understanding. My impression is there are more suitable journal for this paper, such as the EGU journal GMD, or Environmental Modelling and Software.”

First, we would like to highlight the fact that the manuscript has to be considered as a Technical note. While it does not appear here in the title due to technical reasons (the category has been changed after the initial submission), this category is visible when logged in the HESS system. We acknowledge that it could have been misleading to the reviewer and we apologize for that. We also acknowledge that the objective of the paper might be unusual. After discussion with the Chief Editors of HESS, the Technical note format has been identified as the best category for this work. Please note that while HESS Technical notes are supposed to be rather short, an exception to this rule has been proposed by the HESS Chief Editors, as a short format would not have provided sufficient materials to the readers. Finally, other journals have been considered by the authors, namely GMD and EM&S indeed. We chose a Copernicus journal due to the open access format of these journal, which is definitely in line with the openness of the R language. Compared to HESS, GMD is more oriented to large models and we felt that HESS would provide a better reach to the hydrological community, which the HESS Chief Editors agreed with.

We added a couple justifications at the end of the introduction.

"2. The scope of users appears to be restricted to the R users, which is a relatively narrow scope. The authors should think of some more general insights which are useful to the broader hydrological community."

The usefulness of publications is indeed a pre-requisite in many cases (although for some fields or research questions, the usefulness might arise a few years after publication) and providing to the readers a work that could become useful is a clear objective we had in mind when proposing this manuscript. We rather disagree with the reviewer about the fact that R users represent a narrow scope of users. First, the R community of hydrological modellers is quite extensive as demonstrated by Slater et al., (2019) (see for example Figures 1 and 2). Second, rainfall-runoff models are widely used in hydrology, and the models we present in this study are models that are largely used in the world. The compared models cover a large spectrum of potential applications, ranging from applications on snow-influenced catchments to catchments with shallow groundwater as well as for flood and low-flow simulations. Finally, we believe that the number of downloads and views of the preprint so far advocates for a great interest to the hydrological community, with over 1600 downloads from 17 October to 27 November 2020 (see Tab Metrics of https://hess.copernicus.org/preprints/hess-2020-498/) while the most downloaded papers of the last 12 months in HESS start from 2160 views (see https://hess.copernicus.org/most_downloaded_recent.html). In addition, the Slater et al. (2019) paper, which we believe is one reason why we considered the present paper potentially useful to the community, already belongs to the category of the most downloaded papers ever in HESS (in the top 30, see https://hess.copernicus.org/most_downloaded_all_time.html) in spite of being quite recent. Being 'trendy' definitely does not replace scientific and rigorous evaluation by peers, but we find that the elements might indicate that the scope of this paper is not that narrow.

"3. Why R deserves a special attention, is not sufficiently discussed. Is it the case that it is (1) more popular, and (2) more suitable than other programming languages? In terms of popularity, there are other languages often used by hydrologists for modelling purposes, including Matlab, Fortran and Python. In terms of suitability, it should be considered that R, by being an interpreted language, is much slower than compiled languages, or than Python when just in time compiler like Numba are used."

We have chosen the R language because it is very popular for hydrological studies, as demonstrated by Slater et al., (2019). We tried to point it out in the introduction:

- Lines 29 to 32: “The R language is reasonably easy to use and has been taking advantage of a growing community of users. It can be used at each step required for a basic study in hydrology (the hydrological workflow steps, see Fig. 3 of Slater et al., 2019). Consequently, there has been an important increase in the growth and use of hydrological R packages (see Fig. 1 of Slater et al., 2019).”

- Lines 39 to 43: “At a time when data management is a key issue in many branches of science, R has taken a central place in hydrology (Slater et al., 2019). Dealing with the rise of available data can be achieved within the R environment through the numerous packages for data preprocessing, such as `rnrfa` (Vitolo et al., 2016a, 2018) to retrieve hydrological data from the UK National River Flow Archive or `raster` (Hijmans, 2020) to manipulate spatial data.”

As R has a strong community of users, lists of packages related to certain topics are regularly published by the CRAN, and the topic of hydrology was recently dealt with, as said in the mnuscript: “the work of Zipper et al. (2019) who established a list (R Task View) sorting the hydrology-related packages by topics (data retrieval, statistical modelling...). R Task Views are guides — proposed by the CRAN — on the main packages related to a certain topic.” lines 92 to 94.

To further demonstrate the relevance of R for modelling purposes, we propose to add the following at line 29: “A large range of documentation, tutorials, manuals and online discussion platforms are developed by the R-Hydro community, such as what the rOpenSci project (https://ropensci.org) develops or the many code examples available on Stack Overflow (https://stackoverflow.com). Also many short courses and workshops are regularly organised (e.g. the Using R in Hydrology short course at the EGU General Assembly).”

We mentioned in the manuscript that “there are other languages often used by hydrologists for modelling purposes” with examples of available models lines 34 to 38: “A large number can be found on the R platform, such as the HBV model (Bergström, 1976) or TOPMODEL (Beven and Kirby, 1979), while some others are implemented in Python (e.g. EXP-HYDRO hydrological model, Patil and Stieglitz, 2014) or in Matlab with the `MARRMoT` toolbox (Knoben et al., 2019). A significant number of models like MIKE SHE (DHI, 2017) can only be operated through commercial software and platforms.”

In terms of suitability, indeed R is an interpreted language as written on line 26: “For example, the R language (R Core Team, 2020a), which is an open source interpreted language, can significantly fulfill this task.”, which makes it slower than compiled languages: “Computation times tend to be lower when using a compiled programming language rather than using an interpreted one (see Sect. 4.3.2).” lines 589 to 590 and lines 625 to 626 “The results of Fig. 5 show that the packages based on models coded with a compiled programming language have lower CPU times than the packages integrating models coded in R.”. However, in most of the selected packages, the core models are coded in a compiled programming language (see table 8), which resulted in the computation times presented in Fig. 5. Finally, we believe that no language is perfect. Having flaws does not exclude analysing the tools a programming language proposes, especially if this language is largely used. In addition to the living community of R users, R has the advantage of being easy to apprehend, making it rather easy to use for hydrological modellers, who are not necessary highly-skilled developers. In that view, we believe that a specific article on R hydrological modelling packages has an interest.

We added some more explanations in the introduction.

"4. There is not a clear separation between methods and results, but rather, results are presented simultaneously with the methods. The disadvantages of this approach his that (1) there is no clear overview of what and how is presented in the paper, and (2) the methods for comparing such frameworks cannot be easily exported to comparing other frameworks."

Thank you for this comment. We acknowledge that the current version of the manuscript could make the readers to have some difficulties to follow the development of our argumentation and that readers could miss a clear overview of the analysis framework we applied, preventing from easy exportation to other languages. Although we initially thought that the current structure made the paper easier to read, for example to enable readers to pick their topic of interest, we propose the following modification of the structure of the paper. Please note that this proposition could be reconsidered according to the suggestions of the other reviewers and of the associate editor. We propose to add a methodology section before section 2 that would gather:

- previous section 2.1 "Selection of packages";
- beginning of previous section 3 from line 172 to 181;
- beginning of section 3.1 from line 183 to 197;
- beginning of section 3.2 from line 298 to line 300;
- beginning of section 3.3 from line 360 to line 374;
- beginning of section 4 from line 409 to line 418;
- beginning of section 4.1.1 from line 421 to 433;
- beginning of section 4.1.2 from line 489 to 503;
- beginning of section 4.3.1 from line 486 to line 496;
- beginning of section 4.3.2 from line 608 to 624.

The links between explanations in the methodology section and the beginning of each part would be modified accordingly. The outline of this section would therefore be:

2. Methodology

    2.1 Selection of packages

    2.2 Framework for analysing the hydrological models

    2.3 Framework for analysing package practicalities

We added a Methodology section as described above.

"5. There is not a clear separation between modelling decisions such as how to breakup the catchment, which model structure to use in each landscape section, and how to reduce overparameterization and the choice of which package to use. I think the first requires a reflection about concepts, rather than about software packages."

By adding a methodology section before section 2, as we propose in our answer to the Revewier's comment number 4, we think that the justification of the choice of analyses would be clearer. We propose to explain the choice of separating model conceptualization (cf previous section 3) that includes model structure (cf previous section 3.1), how to break up a catchment (cf previous section 3.2), number of parameters, time steps, inputs and outputs required (cf previous section 3.3), from package practicalities in terms of functionalities and usability (cf previous section 4). These functionalities include some ways of avoiding overparameterization:

- Lines 510 to 512 "airGR includes several vignettes, for example on how to estimate parameters within a Bayesian Markov Chain Monte Carlo (MCMC) framework. The WALRUS package is stored on the GitHub platform where a complete set of documents, tutorials and data can be found (e.g. an R script to run a Monte Carlo parameter estimation procedure)."
- Lines 514 to 517: "hydromad offers a vignette and nine demos are available and deal with subjects such as how to estimate the model parameters or how to conduct a sensitivity analysis. Examples of sensitivity analysis and generalised likelihood uncertainty estimation (GLUE) method (Beven and Binley, 1992) are available on the topmodel's website."
- Lines 550 to 553: "Several methods can be used to take uncertainty into account when estimating model parameters. For example, hydromad includes a function to determine feasible parameter sets and estimate prediction quantiles by applying the GLUE method. The FME package (Soetaert and Petzoldt, 2010) enables estimation of parameters within a Bayesian Markov Chain Monte Carlo (MCMC) framework."

However, we think that it is not within the scope of the paper to choose which method is the most relevant to avoid overparameterization.

We introduced the choice of separating models from packages in the Methodology section. However, as discussed in Sect. 6.1, we do not think that it is within the scope of our work to decide upon the appropriate model and package for a specific study. We rather aim to provide all the necessary elements to make this choice.

"6. There is no insight given on the numerical implementation of model equations. Although the authors note this limitation, it is not clear why this topic has been avoided,as it can be quite important."

We thank the reviewer for pointing out that this point is indeed missing. We propose to add this analysis to the section 3.3 of the first version of the manuscript where available time steps are presented. Namely, we propose to add two columns to Table 4 as can be seen below (please note that the columns are only partially filled in at the moment and will be completed later on). The different equations of a hydrological model can be solved using different techniques. The equations are usually solved analytically (the exact solution is determined by integrating the equation for a given time step), explicitly (the solution is approximated by its derivative at the beginning of the time step) or implicitly (the solution is approximated by its derivative at the end of the time step). When the resolution is analytical or explicit, the operator splitting technique is commonly applied to sequentially calculate processes such as evaporation, runoff and percolation (Santos et al., 2018). Numerical solution in hydrology can be seen as part of the mathematical model rather than software implementation, as it changes the results substantially. We will present the different choices made for the models implemented in the packages. Rationales about the importance of consistent numerical resolution of the model equations will be added in the methodology section.

Table 4: Requirements to run the models and types of outputs made available by the packages. D = daily; H = hourly; M = monthly; A = annual; FL = flexible; Num. res. = Numerical resolution; OS = operator splitting; Ana = analytic; Exp = explicit; Imp = implicit; P = precipitation; T = temperature; PET = potential evapotranspiration; DEM = digital elevation model; SA = subbasins area; hypso = hypsometric curve; TS = time series; ET = evapotranspiration; AET = actual evapotranspiration; RC = runoff components; ∽: only some of the time series of runoff components, internal fluxes or store levels are provided. Between parentheses: parameters or inputs of the corresponding snow routine. It is not compulsory to provide the snow routines with the hypsometric curve but it is strongly recommended when enabled by one of the packages. Inputs in italics correspond to static data, i.e. not time series. In this table, for the semi-distributed models, the parameters are considered uniform over the spatial units; in case they are considered distributed, the amount of parameters should be multiplied by the number of spatial units (i.e. HRUs, subbasins...). All the packages return time series of discharge.

Package	Model(s)	Time step(s)	Num. res.	OS	Inputs	Nb. of param.	Outputs					
							TS of AET and TS of RC	TS of internal fluxes		TS of store levels	Spatially distributed	
`airGR`	GR models	H; D; M; A	Ana	✓	P; PET; (T); (*hypso*)	[1 ; 6] (+2)	✓	✓	(✓)	✓	✘	(✓)
`dynatopmodel`	Dynamic TOPMODEL	FL	Imp & Exp		P; PET; *DEM*	8	✓	∽		✓	✓	
`HBV.IANIGLA`	HBV	FL	Exp	✓	P; PET; (T)	[7 ; 9] (+4)	✓	∽	(✓)	✓	✘	(✘)
`hydromad`	GR4J	D	Ana	✓	P; PET	4	✓	∽		✓	✘	
	IHACRES-CMD	FL	Ana	✓	P; PET; (T)	6 (+7)	✓	∽	(✓)	✓	✘	(✘)
	Sacramento	≥ H	Exp	✓	P; PET	13	✓	✘		✓	✘	
`sacsmaR`	Sacramento	D	Exp	✓	P; PET; *SA*; (T); (*hypso*)	13 (+10)	✘	✘	(✘)	✘	✘	(✘)
`topmodel`	TOPMODEL 1995	FL	Exp		P; PET; *DEM*	10	✓	∽		✘	✓	
`TUWmodel`	Modified HBV	≤ D	Exp & Ana	✓	P; PET; (T); (*hypso*)	10 (+5)	✓	∽	(✓)	✓	✓	(✓)
`WALRUS`	WALRUS	FL	Exp	✓	P; PET; (T); *soil type*	3	✓	✓		✓	✘	

We added some information about the numerical resolution of model equations in Sect. 4.3.1. Please note that, following the recommendations of the second Reviewer, the related table has been divided into two distinct tables.

"7. The authors mention how "appropriately using a new model is fundamentally difficult", because of various reason such as software implementations, having to learn new frameworks etc. Many of such difficulties are addressed in so called flexible modelling frameworks, which aim to facilitate model development and systematic comparisons. Such frameworks should be cited as a possible way to go to overcome such difficulties."

Thank you for this comment. We propose to cite these frameworks in the discussion by adding the following sentences: "Flexible modelling frameworks, such as `SuperflexPy` (Dal Molin et al., 2020) and FUSE (Clark et al., 2008), have partially addressed these difficulties. However, their complex conceptualisation makes them more accessible to advanced modellers than to newcomers. The FUSE implementation for R available on GitHub is in need of active maintenance by the community (https://github.com/cvitolo/fuse). The `hydromad` R package provides such a flexible framework to a certain point, as it enables combining different soil moisture accounting functions with several routing modules" after lines 567 to 569: "It is one of the reasons that should encourage users to model within the same framework. Such a framework can be the R environment with the available modelling packages."

Added in Sect. 6.1.

"8. It would be useful to have a comparison of different routing approaches, and how the cell to cell (or HRU to HRU) connections are implemented in different packages."

Thank you for this suggestion. We understand that it would be an analysis of how streamflows are routed from upstream to downstream (sub)catchments. This would be relevant for the packages that integrate models with a degree of spatial distribution, i.e. `dynatopmodel`, `sacsmaR`, `topmodel` and `TUWmodel`. We have partly presented this in the text:

- Lines 334 to 336: "These units are interconnected through the subsurface store updating based on the TOPMODEL theory and produce runoff and baseflow values to generate the final discharge time series (see Fig. 1)."
- Lines 337 to 341: "Dynamic TOPMODEL enables application of other types of HRUs that can be dependent on very different conditions, such as soil properties, land use but also the components of the topographic index. Fluxes between HRUs are controlled by a "flux-distribution" matrix based on the connectivity between the grid squares of the base digital elevation map contributing to the HRUs (for more details see Metcalfe et al., 2015, 2018). This also allows for connectivity between grids within the same HRU."
- Lines 344 to 346: "The HBV model of `TUWmodel` enables a very straightforward spatial configuration where the model is run independently on different zones (with different parameters and inputs), which can be subbasins, elevation zones or any area defined by the user. The discharge outputs from each zone are then summed up based on their relative area to the entire catchment."
- Lines 348 to 350: "The `sacsmaR` package then enables assignment of a different set of parameters to each HRU as well as different data inputs. The water is run upstream to downstream through a hydraulic routing function based on Lohmann et al. (1996)."

We will make this clearer by adding these pieces of information in Table 3.

Added in Table 3.

Response to Reviewer 2

We would like to thank Elena Toth for her very constructive comments on the manuscript and we are glad that she finds that the Technical Note may be of interest for the hydrological community. We agree with most of the comments and suggestions. We provide an answer to each comment and a potential roadmap to change the manuscript accordingly hereafter.

"In particular, I think that the attempt to keep apart but at the same include the differences due to the presence of the snow routines, make some sections much heavier and some tables much more difficult to read 'at a glance', and I would suggest considering to separate all the issues related to such routines (for the models that include it) in a dedicated section (or appendix)."

We believe that the snow routines are integral parts of the hydrological models, which instead of being named rainfall-runoff models should be named precipitation-runoff models, and that they can be very useful to users; snow-dominated basins and snow-impacted basins represent a large part of basins worldwide. For that reason, we built the manuscript with the feeling that discussing snow models in the text was necessary. However, we acknowledge that the description of the snow models might make the reading of the Technical Note a bit more difficult. This difficulty comes for its most part to the models themselves: snow models and rainfall-runoff models represent processes that happen at different time and space scales, making their spatial distribution and input variables, for instance, very different. We think that separating all the elements related to snow calculations from the other comparisons and keeping them in the body of the article would lengthen the article. For all the above reasons, we would prefer to keep the parts related to snow models as they are right now. However, if the associate editor also agrees on the fact that snow models should be presented differently, we would prefer to move most of the related content to the appendices for a better readability. We would keep Table 2 and the explanations of section 2.2 related to the snow routines (lines 113/114, line 149, lines 155/156 and lines 169/170) as they are right now, and add a cross-reference to the appendices. The snow-related section would be indexed as follows:

- Subsection A.1.1, dedicated to the description of the processes present in the snow models, with the explanations of section 3.1 related to the snow routines: lines 198 to 200, lines 212/213, lines 250/251, lines 265 to 267, lines 273/274, line 277, lines 284 to 289;
- Subsection A.1.2, dedicated to the spatial distribution of snow models, with the explanations of section 3.2.1 and Figure 2;
- Subsection A.1.3, dedicated to the requirements of snow models, with a Table for the requirements and outputs related to the snow routines, extracted from Table 4 (see Table A.2 below), and explanations of section 3.3 related to the snow routines: lines 376/377, lines 385/386, lines 392/393;
- Subsection A.2, dedicated to computation times when snow models are used, with the computation times of Figure 5 for the combination of hydrological models and snow routines and the related explanations of subsection 4.3.2: lines 615 to 617, lines 631 to 633.

We would keep the snow rectangles in Figure 1. The last column of Table 5 would also be moved to the appendices.

Table A.2: Requirements to run the snow routines and types of outputs made available by the related packages. P = precipitation; T = temperature; SR = shortwave radiation; hypso = hypsometric curve; TS = time series. Providing the hypsometric curve to the snow routines is not compulsory but it is strongly recommended when enabled by one of the packages.

Package	Model(s)	Inputs		Nb. of param.	Outputs		
		TS	Static		TS of internal fluxes	TS of store levels	Spatially distributed
`airGR`	CemaNeige	P; T	hypso	2	✓	✓	✓
`HBV.IANIGLA`	HBV	P; T		4	✓	✓	✘
`hydromad`	snow.sim	P; T		7	✓	✓	✘
`sacsmaR`	SNOW-17	P; T	hypso	10	✘	✘	✘
`TUWmodel`	HBV	P; T	hypso	5	✓	✓	✓
`WALRUS`	Degree-Day method or	P; T		0	✓	✓	✓
	Shortwave radiation method	P; T; SR		0	✓	✓	✓

Following the suggestions of Reviewer 1, we added a Methodology section. We also added some headers in Sect. 3, Sect. 4.1 and Sect. 5.1, and we separated Table 4 into two distinct tables (now Table 4 and 5). We think that the paper is now much easier to read. Therefore, we chose to keep the analyses related to snow along with the analyses of the hydrological models.

"Introduction: Lines 39-61 may be substantially shortened"

We propose to remove lines 46 to 53 from "In spite of the growing availability" to "comprehensive and controlled protocols.".

Done in the introduction.

Section 2.1: I would have appreciated a wider list (acknowledging, of course, that it cannot be exhaustive) of the R-R models available in R that you have found in your search but have not been chosen (that is, not belonging to the conceptual continuous bucket-type).

Thank you for this comment. We propose to add the following list of packages with the reasons why we chose not to include them in our analysis:

- The `Ecohydmod` package (Souza, 2017) implements an Ecohydrological model.
- The `LWF-BROOK90` package (Schmidt-Walter et al., 2020) implements a physically-based land-surface hydrological model.
- The `fuse` package (Vitolo et al., 2016) proposes a large number of model structure configurations. It was considered that its main purpose was not to conduct a basic hydrological study but more to understand errors arising from hydrological models.
- The `RHMS` package (Arabzadeh and Araghinejad, 2019) implements several event-based hydrological models.
- The `SWATmodel` package (Fuka et al., 2014) implements a watershed hydrological transport model.

Done in Sect. 3 with some more explanations. In addition, we have reorganised the explanations related to our search for different hydrology modelling R packages in Sect. 2.1.

"Section 2.2: I would separate the different models in separate sub-sections (either numbered or just identified with a header), for a better 'indexing"'

Thank you for this suggestion. As some models are included in more than one package, we propose to separate the packages in different sub-sections rather than the models.

Done.

"I would move Table 2 in Section 3.2.1 (or, even better, in separate section/appendix, as suggested above)."

We think that it is more appropriate to put Table 2 before section 3.1.1 (or A.1.1), as we deal with the conceptual storages and fluxes related to the snow routines in this section.

After careful consideration, we find that this table will be of better help in Sect. 3 where we present an overview of the packages, so the readers can look for the appropriate references related to snow routines along with the references related to hydrological models.

“Section 3.1 The legend of Fig. 1 is tiny and, in order to better “read” the figure, I would suggest adding in the text at l. 198 a general description of the meaning of stores and transfers represented in Figure 1. And in the figure, I would add some symbols to the colours (S for snow inside the blue box, P for precipitation arrows, etc), especially needed for the bi-coloured boxes.
I would clarify in this section that details on the input data (input arrows) will be given in section 3.3 and Table 4.
Also here, I would add separate sub-section headers for the different models.”

We will try to improve the readability of Fig. 1 by increasing the size of the legend. We will move the explanations of the caption of Fig. 1 from “The root zone storage” to “on a single spatial unit.” to lines 196/197 after “to ensure consistency.”. We will also add the following sentence just after: “Details on the input data will be given in section 3.3 and Table 4.”. However, we think that adding symbols would overload the diagrams. We will separate the explanations for the different models by adding sub-section headers.

We increased the size of the legend and we moved the explanations of the caption to the beginning of Sect. 4.1. We added the sentence about input data as well.

“Section 3.2.1: It would be necessary to better clarify the differences between the spatial discretization needed by the snow routines (due to the influence of elevation) and the spatial distribution of the models in general (and again, if all the considerations on snow modelling may be moved in a following section/appendix, it would help in better following the main flow of the analysis)”

We acknowledge that more explanations are needed to clarify the differences between the spatial discretization needed by the snow routines and the spatial distribution of the hydrological models. We will try to make this point clearer in the revised manuscript. Depending on where the sections related to snow routines will be placed in the text, we propose to add the following sentences to section 3.2.1 or A.1.2 after line 303 and replacing lines 304/205: “The influence of snow processes on streamflow can vary with elevation, as snow accumulation and melt mainly depend on air temperature that usually decreases with elevation and precipitation that usually increases with elevation. For that reason, a spatial discretization within the catchment may be needed to better account for snow influence when modelling streamflow at the outlet of a catchment. Some packages propose a spatial discretization to account for the influence of snow processes on streamflow. Four configurations were found possible.”

Added in Sect. 4.2.1.

“Section 3.3 Table 4 has too many information and I find it confusing. Again, removing the information on the snow modules, now in the parentheses, would make the table much more readable. Otherwise, I would split it into two tables (one with time steps, input and #parameters and one with the outputs). I would also make two separate columns for the inputs (that is a very important issue for the practical use of the models), distinguishing the meteorological time-series from the static catchment properties (SA, hypso, DEM, etc). In the text, separate (possibly again in sub-sections) the different issues that are discussed: outputs, inputs + number of parameters (since the availability of some static information on the catchment allows to set the values of some parameters), time steps.”

Thank you for this comment. We will separate Table 4 into two distinct tables. If the explanations related to snow routines are kept in the main body of our technical note, the two tables would become Table 4 and 5 as shown below. Otherwise, informations related to snow routines will be moved to section A.1.3.

Table 4: Requirements to run the models and associated numerical resolutions. D = daily; H = hourly; M = monthly; A = annual; FL = flexible; Num. res. = Numerical resolution; OS = operator splitting; Ana = analytic; Exp = explicit; Imp = implicit; P = precipitation; T = temperature; PET = potential evapotranspiration; DEM = digital elevation model; SA = subbasins area; hypso = hypsometric curve; TS = time series. Between parentheses: parameters or inputs of the corresponding snow routine. It is not compulsory to provide the snow routines with the hypsometric curve but it is strongly recommended when enabled by one of the packages. In this table, for the semi-distributed models, the parameters are considered uniform over the spatial units; in case they are considered distributed, the amount of parameters should be multiplied by the number of spatial units (i.e. HRUs, subbasins...).

Package	Model(s)	Time step(s)	Num. res.	OS	Inputs		Nb. of param.
					TS	Static	
`airGR`	GR models	H; D; M; A	Ana	✓	P; PET; (T)	(hypso)	[1 ; 6] (+2)
`dynatopmodel`	Dynamic TOPMODEL	FL	Imp & Exp	✘	P; PET;	DEM	8
`HBV.IANIGLA`	HBV	FL	Exp	✓	P; PET; (T)		[7 ; 9] (+4)
`hydromad`	GR4J	D	Ana	✓	P; PET		4
	IHACRES-CMD	FL	Ana	✓	P; PET; (T)		6 (+7)
	Sacramento	$\geq$ H	Exp	✓	P; PET		13
`sacsmaR`	Sacramento	D	Exp	✓	P; PET; (T)	SA; (hypso)	13 (+10)
`topmodel`	TOPMODEL 1995	FL	Exp	✘	P; PET	DEM	10
`TUWmodel`	Modified HBV	$\leq$ D	Exp & Ana	✓	P; PET; (T)	SA; (hypso)	10 (+5)
`WALRUS`	WALRUS	FL	Exp	✓	P; PET; (T)	soil type	3

Table 5: Model outputs made available by the packages. TS = time series; AET = actual evapotranspiration; RC = runoff components; ∽: only some of the time series of runoff components, internal fluxes or store levels are provided. Between parentheses: outputs of the corresponding snow routine. All the packages return time series of discharge.

Package	Model(s)	Outputs			
		TS of AET and TS of RC	TS of internal fluxes	TS of store levels	Spatially distributed
`airGR`	GR models	✓	✓ (✓)	✓	✘ (✓)
`dynatopmodel`	Dynamic TOPMODEL	✓	∽	✓	✓
`HBV.IANIGLA`	HBV	✓	∽ (✓)	✓	✘ (✘)
`hydromad`	GR4J	✓	∽	✓	✘
	IHACRES-CMD	✓	∽ (✓)	✓	✘ (✘)
	Sacramento	✓	✘	✓	✘
`sacsmaR`	Sacramento	✘	✘ (✘)	✘	✘ (✘)
`topmodel`	TOPMODEL 1995	✓	∽	✘	✓
`TUWmodel`	Modified HBV	✓	∽ (✓)	✓	✓ (✓)
`WALRUS`	WALRUS	✓	✓	✓	✘

We also propose to add the following sub-sections:

- 3.3.1 Inputs and number of adjustable parameters. This sub-section would gather: lines 360/361 from “Since” to “differ”, lines 362/363 from “As data availability” to “requirements”, lines 375 to 386 and lines 391 to 393 from “The two versions” to “snow considerations”.

- 3.3.2 Time steps and numerical resolution of model equations. This sub-section would gather: “The packages allow operation of the models at different time steps and imply different types of numerical resolutions of model equations”, sentences on numerical resolution (see the reply to Reviewer 1) “The different equations of a hydrological model can be solved using different techniques. The equations are usually solved analytically (the exact solution is determined by integrating the equation for a given time step), explicitly (the solution is approximated by its derivative at the beginning of the time step) or implicitly (the solution is approximated by its derivative at the end of the time step). When the resolution is analytical or explicit, the operator splitting technique is commonly applied to sequentially calculate processes such as evaporation, runoff and percolation (Santos et al., 2018). Numerical solution in hydrology can be seen as part of the mathematical model rather than software implementation, as it changes the results substantially" and lines 387 to 391 from "all the packages" to "some constraints".

- 3.3.3 Outputs. This sub-section would gather: lines 363 to 374 from "By making" to "by the packages" and lines 393 to 396 from "While few packages", to "TOPMODEL 1995".

We split Table 4 into two distinct tables (now Table 4 and 5). We added sub-sections in Sect. 4.3.

"Section 4.1.1: it would be better if the text addressed separately each one of the functionalities (sub-sections): preprocessing, calibration, etc."

We will separate the text into the following sub-sections:

- 4.1.1.1 Criteria. This sub-section would gather: lines 435 to 447 from "Weighted combinations" to "see Freer et al., 2004)"
- 4.1.1.2 Parameter estimation. This sub-section would gather: lines 447 to 454 from "Automatic calibration".
- 4.1.1.3 Plot function. This sub-section would gather: lines 455 to 466.
- 4.1.1.4 Graphical user interface. This sub-section would gather: lines 467 to 487.

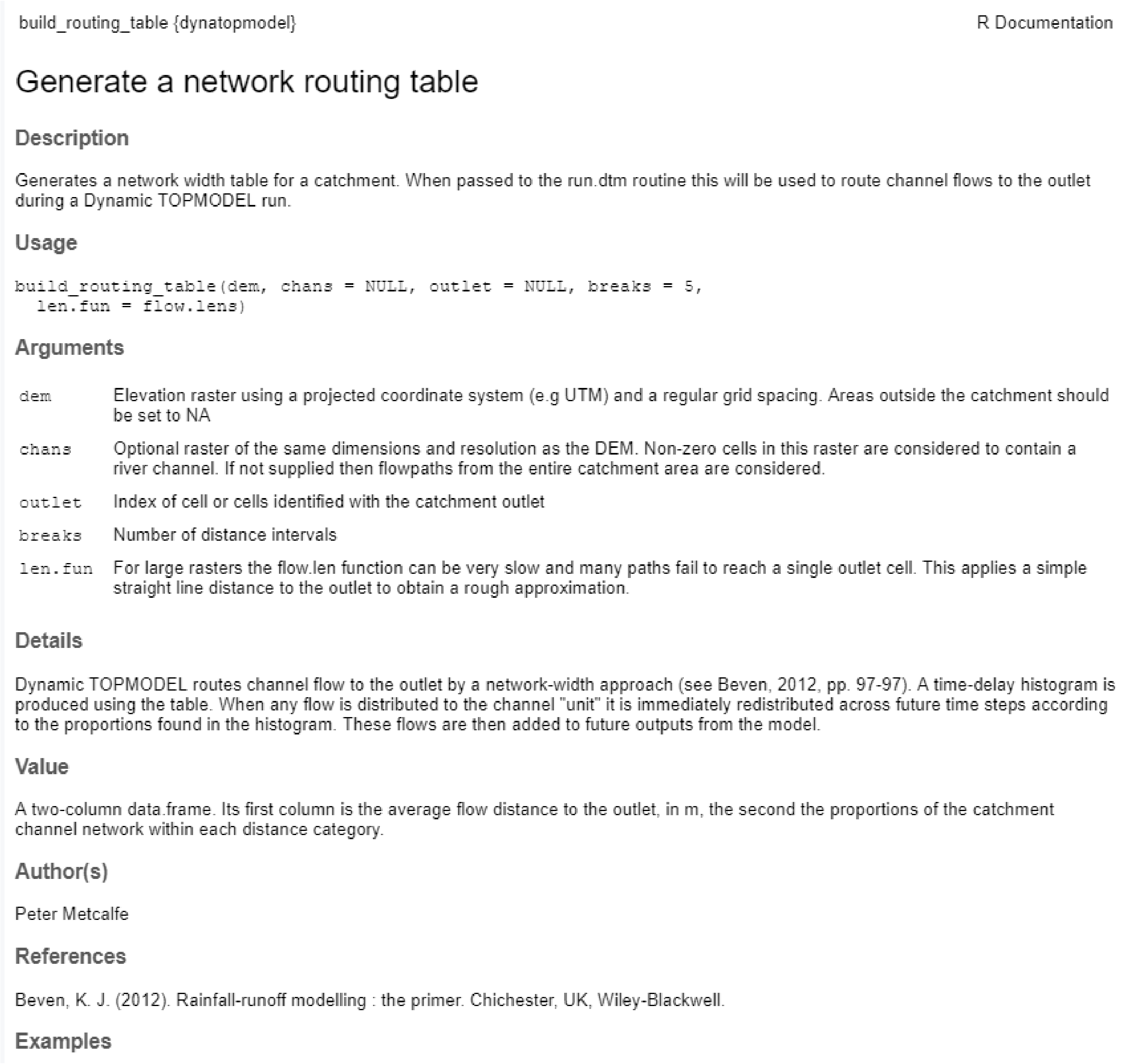
We added headers in Sect. 5.1.1.

"Section 4.1.2 may be substantially shortened, leaving just a few general comments. In fact, Table 6 is not really needed (the distinction of the meaning of different columns is not so sharp). And also Table 7 may be removed, especially since it presents information that may be updated any time, making it out-of-date."

We disagree with this point. We think that Table 6 and 7 summarize important pieces of information for users. Information from Table 6 cannot be accessed by users before installing and manipulating the packages, while information from Table 7 can be difficult to find. Furthermore, it is one of the main points of our work that the documentation of the packages could be improved, which would broaden the use of the packages by making them more accessible to newcomers. From a user perspective, the lack of documentation is one of the limitations when using these packages. In order to make this point, we believe that this analysis must contain some elements of comparison. Also, the differences between the columns of Table 6 correspond to the different fields suggested by R when developing a package (see the example of Fig. 1 page 7 of this document).

"Section 4.2 ll 548-554 may be moved in section 4.1.1 (in the functionalities section)? Fig 4 and text 554-583 should be moved after 4.3.1, since the 'user-function' boxes should be introduced first."

Thank you for this comment. We agree that lines 548-554 should be moved to section 4.1.1. We propose to move these lines to the new sub-section about parameter estimation functionalities. However, we think that Fig. 4 and lines 554 to 583 should stay in section 4.2 as it corresponds to the analysis of the connections between the functions of a package. Therefore, it intends to help users with their implementation of the hydrological workflow. Please note that Fig. 4 should have appeared one page before in the preprint. Also the term "dependency" in the legend of Fig. 4 will be replaced by "Legacy". We propose to add the definition of "user-function" at line 542 after "to run (Fig. 4)".



build_routing_table {dynatopmodel}   R Documentation

Generate a network routing table

Description

Generates a network width table for a catchment. When passed to the run.dtm routine this will be used to route channel flows to the outlet during a Dynamic TOPMODEL run.

Usage

```
build_routing_table(dem, chans = NULL, outlet = NULL, breaks = 5,
  len.fun = flow.lens)
```

Arguments

`dem` Elevation raster using a projected coordinate system (e.g UTM) and a regular grid spacing. Areas outside the catchment should be set to NA

`chans` Optional raster of the same dimensions and resolution as the DEM. Non-zero cells in this raster are considered to contain a river channel. If not supplied then flowpaths from the entire catchment area are considered.

`outlet` Index of cell or cells identified with the catchment outlet

`breaks` Number of distance intervals

`len.fun` For large rasters the flow.len function can be very slow and many paths fail to reach a single outlet cell. This applies a simple straight line distance to the outlet to obtain a rough approximation.

Details

Dynamic TOPMODEL routes channel flow to the outlet by a network-width approach (see Beven, 2012, pp. 97-97). A time-delay histogram is produced using the table. When any flow is distributed to the channel "unit" it is immediately redistributed across future time steps according to the proportions found in the histogram. These flows are then added to future outputs from the model.

Value

A two-column data.frame. Its first column is the average flow distance to the outlet, in m, the second the proportions of the catchment channel network within each distance category.

Author(s)

Peter Metcalfe

References

Beven, K. J. (2012). Rainfall-runoff modelling : the primer. Chichester, UK, Wiley-Blackwell.

Examples

Figure 1: Help page of the `build_routing_table` function included in the `dynatopmodel` package.

Done.

"I find Section 5 too long: I would suggest merging, reorganising and shortening the lines from 664 to 765 (last part of section 5.1 and sections 5.2 and 5.3). It is also important not mixing issues related to the additional functionalities with those related to the documentation. As a user, I believe one of the main obstacles in the use of some R-packages is the fact that the documentation is not modular enough, that is, it does not explain in an immediate way how to do only the specific thing you want to do. And,in this case, this would be for example just running the rainfall-runoff model, without any additional pre- or post- functionality, that I may prefer to carry out autonomously"

Thank you for this comment. We will consider shortening and reorganizing the discussion.

We have not found a way to shorten the discussion without removing some important content. However, for a better readability and structure of the discussions, we proposed some headers.

"Abstract: ll 6-7: I would clarify that some packages include more than one model and the same model may be included in more than one package."

We believe that this information is not necessary in the abstract.

"I would move ll 33-38 (all languages), before ll 25-32 (R, that is one of the possible languages)."

Thank you for this suggestion. We propose the following modifications that also take the comments of the first reviewer into account: "Various types of hydrological models exist according to their assumptions on the representation of natural processes, space and time dependence (e.g. Clark et al., 2011; Beven, 2012; Clark et al., 2017). Various programming languages enable the use of these hydrological models. For example, some models are implemented in Python (e.g. EXP-HYDRO hydrological model, Patil and Stieglitz, 2014) or in Matlab with the MARRMoT toolbox (Knoben et al., 2019). A significant number of models like MIKE SHE (DHI, 2017) can only be operated through commercial software and platforms.

A large number of models can be found on the R platform, such as the HBV model (Bergström, 1976) or TOPMODEL (Beven and Kirby, 1979). The R language (R Core Team, 2020a) is an open source interpreted language. R was originally designed for statistics (as an open source implementation of the S language, Becker et al., 1988) but has since been employed in many other scientific fields. The functionalities of the R language are contained in packages, some of which include features related to hydrology topics. The R language is reasonably easy to use and has been taking advantage of a growing community of users. A large range of documentation, tutorials, manuals and online discussion platforms are developed by the R-Hydro community, such as what the rOpenSci project (https://ropensci.org) develops or the many code examples available on Stack Overflow (https://stackoverflow.com). Also many short courses and workshops are regularly organised (e.g. the Using R in Hydrology short course at the EGU General Assembly). R can be used at each step required for a basic study in hydrology (the hydrological workflow steps, see Fig. 3 of Slater et al., 2019). Consequently, there has been an important increase in the growth and use of hydrological R packages (see Fig. 1 of Slater et al., 2019). Some of these packages are designed for hydrological modelling. In this study, we will restrict ourselves to the hydrological models that are available within the R environment."

Done in the introduction.

"l. 68: I would not use "efficiency" here, to avoid misunderstanding with an assessment of the simulation performance."

We agree. We propose the following modification: "The objective of this paper is to review the pros and cons of using hydrological models implemented as packages in the R environment and to compare and evaluate their applicability and usability."

Done.

"ll. 100-101: not clear."

We propose the following sentence: "We chose to exclude the packages that contain complex models without providing any function for data preparation or any explanatory document (e.g. SWATmodel, Fuka et al., 2014)."

Done along with the list of packages that we have not selected (Sect. 2.1).

"ll 189-194 may be replaced with just the list of references proposing similar diagrams, without explaining in what they differ from yours."

We will remove the explanations on how these legends differ from the one we used. The explanations lines 188 to 195 from "This type of unified" to "from de Boer-Euser et al. (2017, Fig. 3)." would become: "This type of unified comparison was, for example, employed in the framework for understanding structural errors (FUSE, see Fig. 3 of Clark et al., 2008) to compare the structures of four models, or to present the forty models included in the MARRMoT toolbox (see Fig. 2 of Knoben et al., 2019). We selected a different approach for this analysis. The legend we used here was modified from de Boer-Euser et al. (2017, Fig. 3)."

Done.

"l 345: clarify the relation between the basin partition in elevation zones (snow routine) and in subbasins (see comment on snow appendix)."

We propose to replace "The discharge outputs from each zone are then summed up based on their relative area to the entire catchment" at line 346 by: "For example, a catchment can be divided into three subbasins, with one subbasin divided into five elevation zones. The relative contribution of each spatial entity to the entire catchment is defined by the user with a weighting coefficient. The discharge outputs from each zone are then summed up using these coefficients."

Done.

"ll. 352-353: it may be highlighted that the higher flexibility of the spatial discretization of sacsmaR is obtained with a demanding pre-processing to be carried out outside of the package."

We propose to add the following sentence at line 358: "Please note that the high level of spatial discretization enabled by some of the packages sometimes requires a demanding pre-processing to be carried out outside of the corresponding package (e.g. `sacsmaR`, see section 4 and Table 5)."

Done.

"l. 628: can you explain why the HBV.INANIGLA has lower times?"

We have not investigated the reasons that would explain the different CPU times. These reasons can be manifold and imply the choice of programming language, the model complexity, the code implementation and the method of numerical resolution among others.

Response to Mattia Neri

We would like to thank Mattia Neri for his very relevant comments on the manuscript. We provide an answer to each of his comments hereafter.

"Lines 201-204: Please double check the description, "...either a remaining PET component is used to calculate evapotranspiration withdrawn (blue arrow) from the production store (green rectangle) or..." but blue arrow (evaporation) consistently withdraws from the interception storage and green arrow (transpiration) withdraws from root zone/production storage, right?"
The reviewer is right: there is an inconsistency between the text and the figure. The GR4J model's storages and fluxes are conceptual representations of hydrological processes. Here, the remaining PET energy demand after the interception process corresponds to both transpiration and evaporation processes. Therefore, we will add a blue arrow exiting the production store of GR4J on Fig. 1 and change the related explanations by the following sentence at lines 201/202: "Then, either a remaining PET component is used to calculate evapotranspiration withdrawn (blue and green arrows) from the production store (green rectangle) . . . ".
Done.

"Lines 312-316: I think that TUWmodel elevation zones can be set by the user as de-sired, even using different ranges if needed (if inputs and drainage areas are defined accordingly). Moreover, I would specify that, in contrast to CemaNeige, model parameters could be differentiated across elevation zones."
Thank you for this comment. Indeed, TUWmodel enables to set the elevation zones with different ranges if needed. We will replace the explanations at lines 312-319 by: "The spatial distribution of snow processes by the TUWmodel and sacsmaR packages follow another principle, the difference being that the elevation zones can be set with different ranges and with different surface areas (e.g. Fig. 2). Model parameters can be differentiated across elevation zones". We will also correct Fig. 2 (see Fig. 2 on page 2 of this document). As stated lines 344-346: "The HBV model of TUWmodel enables a very straightforward spatial configuration where the model is run independently on different zones (with different parameters and inputs), which can be subbasins, elevation zones or any area defined by the user", the parameters of TUWmodel can be differentiated across elevation zones. We will add this explanation in the section related to the spatial discretization of snow modules (section 3.2.1).
Done.

"Table 4: I think that TUWmodel requires also subbasin areas."
Thank you for this comment. TUWmodel indeed requires subbasin areas when used in a semi-distributed mode. We will add this information in Table 4.
Done.

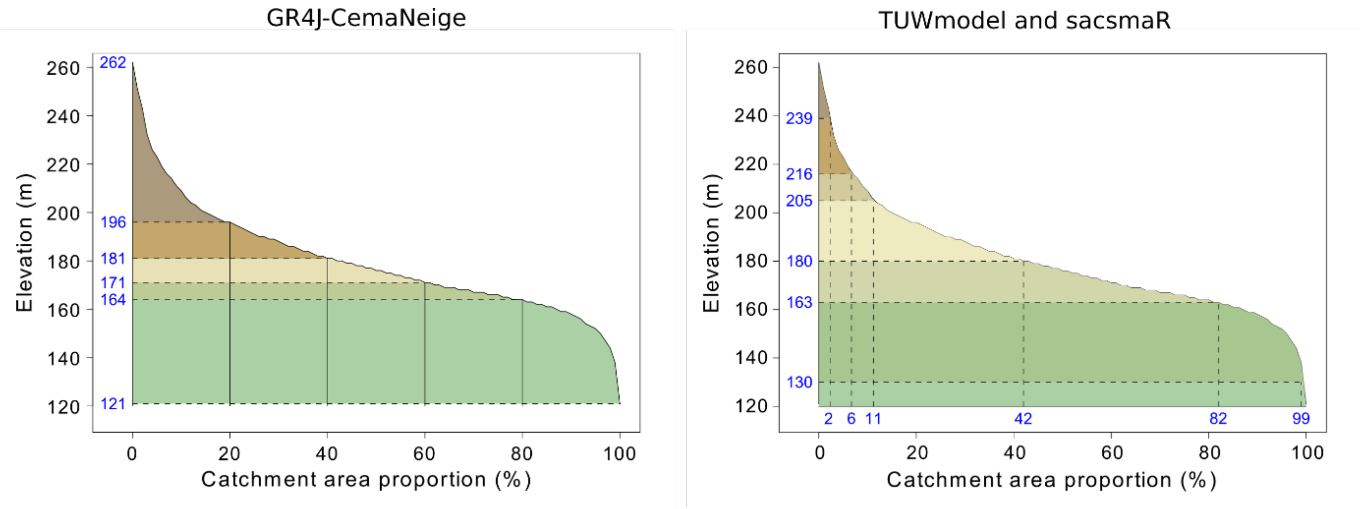


Figure 2: GR4J-CemaNeige elevation zones (left) and TUWmodel/sacsmaR elevation zones (right) both following the hypsometric curve of the Couetron River at Souday (France). Each colour indicates a different elevation zone.

"Section 4: The authors analyse several package functionalities, as the presence of an automatic calibration function for the hydrological models. However, for the packages which do not include calibration functions (but, in general, for all packages if the user wants to implement a different calibration procedure) is there any advice on the suggested parameter ranges or any kind of related information? I would add such specific information along with the calibration functionalities: it is indeed helpful to the user for the implementation of the models."

We have provided several references (see section 2.2 and Table 1 and 2) that could help users to find possible ranges of parameters for different modelling applications. However, we think that it is not within the scope of this Technical note to provide guidelines on the most appropriate range of parameters for a given model. Indeed, these recommendations would depend on the modelling objectives, the chosen strategy for parameter estimation and several other modelling aspects.

"Lines 434-438: please clarify, e.g. what do you mean with "combination of criteria" for preprocessing?"

Thank you for this comment. The combination of criteria is not related to preprocessing of input data but to the calculation of performance (or error) criteria. We acknowledge that the current version of the text might be confusing and we will slightly modify it to make it clearer. In addition, we believe that adding sub-headers to section 4.1.1 will increase the readability of this part. A combination of several criteria can be, for instance, a weighted sum of three criteria. For example, in airGR, users can average the KGE calculated on discharge, the KGE calculated on square root discharge and the KGE calculated on the inverse of discharge, and different weights can be chosen for each of these three individual criteria.

We added a clearer definition of "combination of criteria" in Sect. 5.1.1.

References

Arabzadeh, R. and Araghinejad, S.: RHMS: Hydrologic Modelling System for R Users, URL https://CRAN.R-project.org/package=RHMS, R package version 1.6, 2019.

Beven, K. J. and Freer, J.: A dynamic TOPMODEL, Hydrological Processes, 15, 1993–2011, https://doi.org/10.1002/hyp.252, 2001.

Clark, M. P., Slater, A. G., Rupp, D. E., Woods, R. A., Vrugt, J. A., Gupta, H. V., Wagener, T., and Hay, L. E.: Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models, Water Resources Research, 44, https://doi.org/10.1029/2007WR006735, 2008.

Dal Molin, M., Fenicia, F., and Kavetski, D.: SuperflexPy: the flexible language of hydrological modelling, URL https://superflexpy.readthedocs.io/en/latest/index.html, version 1.2.0, 2020.

Fuka, D. R., Walter, M. T., Steenhuis, T. S., and Easton, Z. M.: SWATmodel: A multi-OS implementation of the TAMU SWAT model, URL https://CRAN.R-project.org/package=SWATmodel, R package version 0.5.9, 2014.

Santos, L., Thirel, G., and Perrin, C.: Continuous state-space representation of a bucket-type rainfall-runoff model: a case study with the GR4 model using state-space GR4 (version 1.0), Geoscientific Model Development, 11, 1591–1605, https://doi.org/10.5194/gmd-11-1591-2018, 2018.

Schmidt-Walter, P., Trotsiuk, V., Meusburger, K., Zacios, M., and Meesenburg, H.: Advancing simulations of water fluxes, soil moisture and drought stress by using the LWF-Brook90 hydrological model in R, Agricultural and Forest Meteorology, 291, 108 023, https://doi.org/10.1016/j.agrformet.2020.108023, 2020.

Souza, R.: Ecohydmod: Ecohydrological Modelling, URL https://CRAN.R-project.org/package=Ecohydmod, R package version 1.0.0, 2017.

Vitolo, C., Wells, P., Dobias, M., and Buytaert, W.: fuse: An R package for ensemble Hydrological Modelling, The Journal of Open Source Software, 1, https://doi.org/10.21105/joss.00052, 2016.