Reply to Anonymous Referee 1

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

Comment:

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.

Reply:

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.

Comment:

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.

Reply:

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.

<u>Comment:</u>

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.

<u>Reply:</u>

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.

Comment:

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.

Reply:

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

<u>Comment:</u>

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.

<u>Reply:</u>

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.

<u>Comment:</u>

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.

<u>Reply:</u>

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	✓	✓	✓	✗

Comment:

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.

Reply:

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."

Comment:

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.

Reply:

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.

References

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://superf lexpy.readthedocs.io/en/latest/index.html, version 1.2.0, 2020.

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.