1 General comments

Summary The manuscript *A distributed simple dynamical systems approach (dS2 v1.0) for computationally efficient hydrological modelling* by Buitink and colleagues introduces a distributed hydrological conceptual model oriented at the simple dynamical systems approach by Kirchner (2009). The model has been designed to simulate rainfall–runoff dynamics with high spatial and temporal resolution at small (10^0 km^2) up to mesoscale (10^3 km^2) catchments. The model should be straightforward to apply, computationally efficient, and the associated Python code is openly available and easy to understand and modify. In their manuscript the authors present a sensitivity analysis of model parameters and apply the model in the alpine Thur catchment.

Most parts of the paper are well structured and good to understand. However, I see some serious problems with the paper (and the model) that should be resolved before final publication. Even more than for my concerns regarding originality and efficiency this holds true for the applicability of the model, as I will explain in the following paragraphs.

Originality As even the authors admit, many different conceptual hydrological models already exist. Therefore the question arises if we really need yet another model. My personal answer would be: maybe, if we can learn something from it. However, dS2 seems to me rather like a typical black box model and I don't see a way to learn anything about processes or the catchment. Even though the authors see their model as a valuable tool for educational purposes (page 20, line 17 ff.), other conceptual models based on multiple storages (or buckets) are, in my opinion, much more suitable for education. For instance, storages can be related to natural phenomena (e.g. surface, quick subsurface, and baseflow) and students can learn something about processes and parameter sensitivities (for an illustration, see e.g. Fenicia et al., 2016). As far as I can tell, dS2 serves solely as a very simple rainfall–runoff transformation tool and is therefore nothing new. I wonder how we can learn something about processes in a catchment, as the authors claim. To me this is not evident from the example application.

Efficiency The authors stress at many occasions the computational efficiency of their model. On the other hand, computational efficiency has been sacrificed in favour of code readability by implementing it in Python, a widely used scripting language that is known for its well structured syntax in contrast to much more efficient compiled languages such as C++ or Fortran. Therefore, in my view the argument of computational efficiency, which is even advertised in the title of the manuscript, does not hold. A good compromise would have been to outsource the most expensive parts of code to a compiled language and use Python as an interface, as has been done for other models (e.g. TOPMODEL, for which an R interface exists, while the core model is written in Fortran, see Metcalfe, Beven and Freer, 2015).

Applicability I acknowledge that the code of the model is indeed well readable, even for persons with little Python experience (including me). I only wonder how I should apply the model. The authors present an example application in their manuscript but no example data to test the model. What is even more, the input and output file structures are not well explained, neither in their submitted manuscript nor in the assets or the github repository. In that way I don't know how to initialise and apply the model. This is probably the most severe flaw of the presented work. In my opinion it is also violating the core principles of GMD's model code and data policy.

Sensitivity analysis I see a problem with the Sobol' method for variance-based sensitivity analysis that the authors applied. This method may not be applied in case parameters are correlated (which is the case, as is stated in the text). I will provide some more reasoning and suggestions in the specific comments.

2 Specific comments

In the following I will provide some specific comments. Quotations from the paper appear as *emphasised text*. Specific passages in the text are referenced as pmLn, meaning page m, line n.

- *"There is a growing need for easy-to-apply models that can utilize the potential of spatially distributed input data"* (p2L26–27). Many (most?) models already utilize spatially distributed information. I also think the demand for *easy-to-apply* models always has been present (whereas the judgement of what is *easy-to-apply* is rather subjective).

- *"water is most often transferred to the outlet as a post-processing function. This is, however, not necessarily the most computationally efficient way to deal with spatially distributed data"* (p2L29–30). And what is the most (or at least more) efficient approach? Isn't the dS2 model essentially doing the same or what exactly is the difference?

- *"Many aspects of distributed modelling [...] require a high number of runs which further increases the computational demand. An efficient distributed conceptual model to tackle these kind of issues is currently lacking."* (p2L34–35). With computational infrastructure nowadays (HPCs, cloud computing) it is already possible to conduct hundred thousands or even millions of iterations of conceptual models within acceptable time frames, depending on resolution and spatio-temporal domain (see examples in Beven and Binley, 2014). Why then do we need more computationally efficient *conceptual* models? To conduct billions of runs in the same time? I doubt that this would substantially increase the value of uncertainty analysis and parameter estimation.

- *"Low-level languages generally perform faster calculations, but this comes at the price of user-friendliness and ease-of-use."* (p9L3–4). I don't understand this relationship.

Doesn't user-friendliness and ease-of-use depend on the interface between model and user, i.e. the structure of input and output files, how to run the model etc.? You can also program complicated models with high-level languages. And what about users, who prefer GUIs? For them a purely command line-based model will never be considered user-friendly. Moreover, user-friendly models, in my opinion, always contain a good explanation of file structures and a simple test case, which is unfortunately not the case for dS2.

- *"Simulating Europe for three months at hourly time steps and at a resolution of 5x5 km2"* (p9L18). Does it make sense to do that with dS2? Otherwise I don't understand why you choose such an unrealistic comparison.

- Section 3.2 (Adaptive time stepping): I think this sub-section can be hard to follow for anyone who is not familiar with numerical integration. E.g. what is the *Runge-Kutta scheme*, what are *fourth and fifth order estimations*? Maybe you can add a paragraph with a (very) short introduction to numerical integration, how it basically works, and why it is needed. It might also be worthwhile to stress the relevance in hydrological modelling as the issue is often neglected (most hydrological models just use explicit Euler with operator splitting). Some interesting papers about the topic: Clark and Kavetski (2010), Kavetski and Clark (2010), Kavetski and Clark (2011) and Schoups et al. (2010).

- Comment on numerical integration: As the dS2 is so extremely computationally efficient I wonder why you don't try the much more accurate implicit solvers. Of course they will increase computation time, but on the other hand will deliver much more accurate results (and potentially more reliable parameter and uncertainty estimations, see e.g. Kavetski and Clark, 2011). Maybe the GNU Scientific Library (`https://www.gnu.org/software/gsl/`) is worth a try (never tried by myself, but the website says *the interface was designed to be simple to link into very high-level languages, such as GNU Guile or Python*).

- *"[...] base the volume estimation on the mean discharge of the resulting shorter steps"* (p13L2). Is that indicated by $Q_{internal}$ in Fig. 7? Please clarify that in the figure, as $Q_{internal}$ itself is not explicitly defined.

- *"the current version of the model only outputs discharge at the end of the time step"* (p13L8–9). But internally $Q_{internal}$ is used, i.e. $\int_{t-1}^{t} Q_{internal}$ to calculate S_t?

- Fig. 8: In the text you write *response of the model to each parameter*, but the model contains more than the five parameters shown in Fig. 8?! Please be more explicit about what you mean.

- Section 4 (Parameter sensitivity → Sobol' sensitivity analysis): The Sobol' method requires that parameters are independent but in relation to Fig. 8 it is mentioned that some parameters are correlated. This will distort the results of the sensitivity analysis. Or produce strange results as can be seen in Fig. 8, namely that in some cases the total effect is smaller than the main effect (or does the size of the bars reflect main + total effect? Please clarify). It seems also strange that for $KGE - \beta$ the total effect is always zero (if my interpretation of the bars is right). In any case, a possible workaround for correlated parameters is presented by Kucherenko, Tarantola and Annoni (2012) (not really an ad-hoc implementation). However, to see if the effort is really necessary, please first check the correlations among parameters (e.g. via covariance matrix). You might also consider a different method for sensitivity analysis, which is not affected by correlations (see review papers for guidelines, e.g. Pianosi et al., 2016).

- *"These graphs also show that there are some parameter correlations influencing the results"* (p16L1). What exactly do you mean? The discrepancy between main and total effects I mentioned earlier?
- *"We see that, although the magnitude of the peak is not fully captured in the Rietholzbach and Andelfingen basins, the timing of the peak is well simulated in all three basins."* (p17L8–9). But for Andelfingen the simulated peak occurs several hours before the measured peak, even with the routing module (the difference between routing and no routing module seems to be less than the difference between observations and routing module). As I see, this specific issue has also been addressed by the other reviewer.

3 Technical corrections

p1L1	efficiency → efficient
p1L8–9	*"at high temporal and spatial resolution"*: one of the two occurrences of that phrase can be scratched.
p3L23	*"in a flexible and efficient manner"*: also mentioned twice in the same sentence.
p10L11	*"Since this concept [...]"*: As here a new sub-section starts, please specify explicitly what is meant by *this concept*. I.e. the dS2 model concept or the concept of adaptive time stepping (as this is the heading of the sub-section)?
Fig. 6	I think this figure is rather hard to understand (also because of the two meanings of the Y axis). In panel 1) what do the + signs represent?
Figures	All panels must be indicated with brackets around lower case letters, e.g. (a), (b). See house standards (the proof reading will care about that anyway).

p14L1	*"It is striking that the three g(Q) parameters [...]"*: a short reminder what the three parameters are (of the four shown in Fig. 8) would be helpful.
p14L8	*"due to the large equifinality regions"*: it would be good to help readers with the interpretation, e.g. by adding to the mentioned sentence something like *"shown in dark blue"*.
p14L9	full stop.
Text in general	Note the difference between different dashes, i.e. '-' and '–' (and sometimes '—'). I think throughout the text always '-' is used, where it is sometimes not appropriate, e.g. it should read *Kling–Gupta efficiency* (instead of *Kling-Gupta*), see GMD guidelines.
Fig. 10	Please describe in the caption, what the KGE value in brackets stands for and what are the *"selected best runs"* (figure should be self-explaining).

References

Beven, Keith and Andrew Binley (2014). 'GLUE: 20 years on'. In: *Hydrological Processes* 28.24, pp. 5897–5918. DOI: `10.1002/hyp.10082`.

Clark, Martyn P. and Dmitri Kavetski (2010). 'Ancient numerical daemons of conceptual hydrological modeling: 1. Fidelity and efficiency of time stepping schemes'. In: *Water Resources Research* 46.10, W10510. DOI: `10.1029/2009WR008894`.

Fenicia, Fabrizio et al. (2016). 'From spatially variable streamflow to distributed hydrological models: Analysis of key modeling decisions'. In: *Water Resources Research* 52.2, pp. 954–989. DOI: `10.1002/2015WR017398`.

Kavetski, Dmitri and Martyn P. Clark (2010). 'Ancient numerical daemons of conceptual hydrological modeling: 2. Impact of time stepping schemes on model analysis and prediction'. In: *Water Resources Research* 46.10, W10511. DOI: `10.1029/2009WR008896`.

— (2011). 'Numerical troubles in conceptual hydrology: Approximations, absurdities and impact on hypothesis testing'. In: *Hydrological Processes* 25.4, pp. 661–670. DOI: `10.1002/hyp.7899`.

Kucherenko, S., S. Tarantola and P. Annoni (2012). 'Estimation of global sensitivity indices for models with dependent variables'. In: *Computer Physics Communications* 183.4, pp. 937–946. DOI: `10.1016/j.cpc.2011.12.020`.

Metcalfe, Peter, Keith Beven and Jim Freer (2015). 'Dynamic TOPMODEL: A new implementation in R and its sensitivity to time and space steps'. In: *Environmental Modelling & Software* 72, pp. 155–172. DOI: `10.1016/j.envsoft.2015.06.010`.

Pianosi, Francesca et al. (2016). 'Sensitivity analysis of environmental models: A systematic review with practical workflow'. In: *Environmental Modelling & Software* 79, pp. 214–232. DOI: 10.1016/j.envsoft.2016.02.008.

Schoups, G. et al. (2010). 'Corruption of accuracy and efficiency of Markov chain Monte Carlo simulation by inaccurate numerical implementation of conceptual hydrologic models'. In: *Water Resources Research* 46.10, W10530. DOI: 10.1029/2009WR008648.