Editor#

Dear Song et al.,
First of all, I would like to commend the authors for this interesting work, which aligns well with the scope of HESS. Furthermore, I would like to underscore the great discussion with both reviewers. In your last response to Dr. Ehret, you outlined a promising strategy for enhancing your manuscript, and I eagerly anticipate a revised and more streamlined version.

Two minor comments to augment the reviewers' feedback:

1. The term "outperform" and the pronounced emphasis on model comparison appear somewhat misaligned with your model's results and also seem to occupy excessive space in this manuscript. How much does your model's performance vary by merely re-training with an alternative weight initialization method, adjusting your hyperparameters, or by interchanging the training and testing datasets? My intention is constructive; I am not suggesting further tests but rather expressing that the approach and concept itself are intriguing and novel. I would have appreciated a deeper explanation and discussion on how this work could be extended to problems beyond bucket models or / and a nice schematic figure with some extra text how this is implemented in the network. Particular because your code is not shared yet which seems unnecessary because the preprint in HESS is open.

2. "Since daily forcing and streamflow data are readily available and accessible, while hourly input data is often more challenging to acquire, our study focuses exclusively on daily hydrological modeling using conceptual models." A quick Google search (10.5281/zenodo.4072700) revealed the existence of datasets (as well as a API of the USGS), and it's likely there are more I am unaware of. This is merely a suggestion should you wish to conduct your experiments at an hourly resolution. I do not think these experiments are necessary here (although interesting); however, I found the argument regarding the lack of data to be somewhat tenuous.

Again congratulations on the nice work and I look forward to the revised manuscript.

Sincerely,

Ralf Loritz

Dear editor,

Thank you for your comments. The effects of the weight initialization method (random number selection) or adjusting hyperparameters are negligible, as this work has been done by Feng et al., 2022, regarding the sequential model. Our adjoint model inherited their hyperparameter configuration to ensure a fair comparison and we did not attempt hyperparameter tuning. Limited tuning suggests the results to be stable. We have added Figure 2 to provide a schematic representation of how the adjoint is utilized within the code for better understanding. The code for the adjoint differentiable models has updated as an attachment for reviewers.

We have revised the manuscript based on the responses proposed in our discussion with the two reviewers. We appreciate their valuable feedback, which prompted us to further consider the coordination between the numerical schemes, the temporal resolution of forcing, and the parameter learning functions (neural networks) in the differentiable hydrological models. We conducted tests on the differentiable HBV model with various numerical schemes and fixed smaller time steps (Table A2 & Figure A3 in Appendix) We added the following main points in our discussion section:

- Directly training an hourly model with ML techniques remains computationally expensive and may lead to the issue of gradient vanishing if the time step is too small, according to the discussion in Gauch et al. (2021). In the literature, some ML techniques have been used to predict hourly flood hydrographs using daily flow data, which require further investigation in the differentiable models.
- Our new results added to the Appendix shows the sequential model and implicit adjoint model with a 1-day time step have higher performance than the explicit Euler schemes with smaller time steps or the fourth-order Runge-Kutta scheme when using daily inputs.
- The forcing and physical parameter configuration in the differentiable model might need to match the timesteps of the numerical schemes to reflect the hydrograph changes within a day if smaller timesteps are used.
- The adjoint used in this work is derived for the gradient of the Newton solver, such that it can theoretically support any model that can be solved with the Newton solver—not only bucket models governed by ODEs but also distributed models governed by PDEs. However, the challenge may still exist in calculating the Jacobian matrix for a batch of basins for PDEs, as the distributed parameters in PDEs are significantly greater in number than those in ODEs, and can slow down the efficiency of the model in both forward and backward modes.

Thank you and the reviewers for your constructive feedback!

Chaopeng

Reviewer #1

Dear authors, Dear editor,

Here is my review of the submitted work. I recommend accepting the manuscript after minor revisions.

Kindly

Ilhan Özgen-Xian

General comments and questions

1. The authors convincingly make an argument for implicit time integration. The forward Euler time stepping used in this work is indeed at a disadvantage if fixed time steps are used. However, it is not clear to me how higher order explicit time integration methods such as schemes from the explicit Runge-Kutta family (RK) would perform in comparison to the implicit one. If I understood correctly, some of the numerical issues mentioned in the manuscript might also be addressed by (adaptive) multistep schemes of this type. The

advantage of RK-type schemes in this context is that the number of computations per time step is known a priori. In contrast, the Newton-Raphson iterative solver may require any number of steps until convergence. High order RK schemes, for example the standard RK45 or the adaptive RK-Fehlberg method, could also potentially benefit from the adjoint method presented in this paper to avoid excessive memory usage. Perhaps the authors can comment on this.
2. The authors mention that the Newton-Raphson solver introduces some overhead to the computation. On average, in the results shown in this paper, how many iteration steps were necessary for the solver to converge?

Thank you for your suggestions. We believe these two questions can be addressed together. The Newton-Raphson solver typically converges in 3-4 iterations on average (information added in line 611). The number of iterations or steps in the Newton solver and the RK scheme are comparable. The primary computational burden of the implicit scheme lies in calculating the Jacobian matrix for the Newton solver, whereas explicit schemes only require forwarding through the physical model. We tested both the 4th order RK scheme and the explicit Euler schemes with fixed, smaller time steps (4 hours and 1 hour). Their results are reasonable but not as robust as the sequential model and implicit adjoint model with one day time step we reported in the main text (please refer to Table A2 and Figure A3 in the Appendix which shows that their metrics are not better than the sequential and implicit models). The most likely reason is that the daily forcing inputs and physical parameters from the neural network do not align with the smaller time steps within a day. The neural network for parameterization is configured to match the temporal resolution of the input data, which is one day. Consequently, the forcing and physical parameters remain constant within a day, failing to capture diurnal changes in forcing. Using explicit solvers with smaller time steps within the differentiable model framework needs to be coordinated with modifications to the forcing inputs and training target data. However, even though hourly data are now publicly available, directly training an hourly model using ML techniques are more computationally expensive and may lead to the well-known issue of gradient vanishing (Gauch et al., 2021). Some ML techniques, such as multi-time-scale learning, have been considered for converting daily flow data into hourly flood hydrographs (Gauch et al., 2021; Sarıgöl and Katipoğlu, 2023). We have an ongoing study working on this topic.

We will add some explanations in our discussion: *"While this work focuses on enabling implicit solvers in differentiable modeling, we do not suggest that explicit solvers are to be discouraged. It has long been explored in the numerical algorithm literature that each type of solver has advantages and disadvantages and is suitable for different problems. For example, implicit solvers are not only preferred but also necessary for stiff ODEs, especially those with dynamics on vastly different time scales and those resulting from the discretization of elliptic PDEs. Using explicit solvers for them could necessitate very small time steps.*

Further complications of using explicit schemes with small time steps include computational expenses, parallel efficiency, and matching forcing functions. Even though hourly data are now publicly available, directly training an hourly model with ML techniques remains computationally expensive and may also cause the notorious problem of gradient vanishing if the training time steps are too numerous (Gauch et al., 2021; Greff et al., 2017). The numerical schemes employed in the physical models within the differentiable modeling framework need to maintain stability for simultaneous large-scale simulations in each minibatch while also allowing for gradient tracking.

Batched learning and parallel efficiency may prefer uniform operations across basins and challenge the application of adaptive time-stepping algorithms. We conducted tests on the differentiable HBV model with various numerical schemes and fixed smaller time steps (Table A2 & Figure A3 in Appendix). The sequential model and implicit adjoint model with a 1-day time step presented higher performance than the explicit Euler schemes with smaller time steps or the fourth-order Runge-Kutta scheme. The main reason may be that the daily forcing inputs and daily physical parameters from the neural network do not match the smaller time steps within a day. Thus, explicit schemes with smaller time steps may require matching forcing functions as well. Some multi-time-scale ML techniques have been used to predict hourly flood hydrographs using daily flow data to avoid gradient vanishing issues in the direct hourly training (Gauch et al., 2021; Sarıgöl and Katipoğlu, 2023). These approaches present possible solutions for future investigations." Line {570-590}

Gauch, M., Kratzert, F., Klotz, D., Nearing, G., Lin, J., Hochreiter, S., 2021. Rainfall–runoff prediction at multiple timescales with a single Long Short-Term Memory network. Hydrol. Earth Syst. Sci. 25, 2045–2062. https://doi.org/10.5194/hess-25-2045-2021

Sarıgöl, M., Katipoğlu, O.M., 2023. Estimation of hourly flood hydrograph from daily flows using machine learning techniques in the Büyük Menderes River. Nat. Hazards 119, 1461–1477. https://doi.org/10.1007/s11069-023-06156-x

Minor comments

1. P.2, L.70: "graphical processing units" should be "graphics processing units"

Revised as suggested.

2. P.3, LL.105ff.: Does "elliptic operator" in this context correspond to the Laplacian? If so, some of the examples might require some annotation. The Saint-Venant equation only contains Laplacian operators if molecular/turbulent diffusion is accounted for. Many forms of the Saint-Venant equation omit these terms, for example (García-Navarro et al., 2019, doi:10.1007/s10652-018-09657-7; LeVeque et al., 2011, doi:10.1017/S0962492911000043).

We revised it to "shallow water equations", which refers to a two-dimensional Saint-Venant equation considering turbulent diffusion.

3. P.3, LL.105ff. (continued) When I looked at the paper by Aboelyazeed et al. (2023) (cited by the authors), I couldn't see Laplacians in the Farquhar model equations.

Aboelyazeed et al. (2023) is an example of systems of nonlinear equations, not an elliptic operator.

4. P6, L.209: "The same forcings ... was used" should be "The same forcings ... were used"

Revised as suggested.

5. P.12, L.335: Should it be Eq. (28) instead of Eq. (27)? May be I am misunderstanding something.

Yes, your understanding is correct. We fixed it.

6. P.14, L.398: The authors state that the mass balance preservation of the adjoint-driven NN-HBV model might be the reason behind the improved model performance. I don't understand why the mass conservation should significantly differ from the explicit sequential NN-HBV model if the hydrological process representation remains untouched. Is this related to the use of thresholds to avoid negative storages? Can the authors elaborate a bit more?

Yes, by avoiding thresholds for negative states, the implicit model can achieve better mass conservation. This impact is significant for low flows. For example, the thresholds for lower subsurface zone storage in the current HBV model can induce minimal baseflow on dry days. More importantly, the adjoint (implicit) model greatly reduces numerical errors, thus improving the model's performance. We revised the main text to: *"The advance may be attributable to HBV.adj's reduction of numerical errors, which forces the model to more accurately represent extreme values."*

7. P.24, L.580: The additional computational cost introduced by the implicit solver is quite substantial (18 h vs. 133 h), suggesting either poor convergence or large communication overhead in the implicit scheme.

We agree with the reviewer that the computational cost of the implicit solver is substantial compared to the sequential model. However, when compared with traditional models that require basin-by-basin calibration on a CPU, it is efficient for large-scale modeling. The implicit solver can converge in 3-4 iterations, and all training is conducted on a single GPU, with no need for communication between nodes. The reasons it is slower than the sequential model are twofold: 1) the HBV model is called 3-4 times in each time step, whereas the sequential model only needs to call it once, and 2) the calculation of the Jacobian matrix for multiple basins, depending on the batch size, also consumes time. There are additionally some CPU overhead issues to be explored down the road.

Reviewer #2

Dear Chaopeng Shen, dear Authors, dear Editor,

Thanks for your detailed replies to my review. This is exactly what the HESS open discussion forum is made for. First of all, I would like to state once more that I think the authors, by introducing in their manuscript a method for integrating implicit solvers in modern ML-based model training via backpropagation, provide valuable research that is absolutely worth publishing. The reason I was (and am) recommending "reject with strong encouragement for resubmission" is that I suggest changes to the manuscript that will most likely require more time than usually assigned for major revisions. I will leave it to the Editor whether he thinks i) the changes I suggest are valid, and if yes, ii) how much time they would require. Reading the replies (AC2, text and .pdf, and AC3) by Chaopeng Shen, four main points of discussion arise. I reply to them here in summarized form, rather than individually in each document:

- ***The main purpose of the paper is to enable implicit schemes.*** *I agree with this statement by Chaopeng Shen, and I suggest focusing on this message. Therefore, I would support a revised paper, as a technical note, that introduces the method, not more and not less. In such a paper, there is no need for an in-depth discussion about if and when explicit schemes for conceptual hydrological models operating on daily data create problems, and no need for comparing implicit schemes vs. explicit schemes operated on higher-resolution data comparison.*
- ***Running small time steps (less than a day) with automatic differentiation creates problems (memory use, allowable window size).*** *Chaopeng Shen suggests it is a bit unfair by me to ask the authors to demonstrate that implicit schemes solve a problem of explicit schemes. I see two options here. The first is to write a short technical note presenting only the main method innovation, see previous bullet point. The second is to keep it as a research paper, showing the method innovation and applications. In that case, as a reader I would expect a demonstration that the innovation solves an important problem present in the applications used in the paper (hydrological modeling on daily basis using conceptual models). That is, showing that the currently used explicit schemes introduce substantial numerical error. This does not need to be for the full set of catchments used, but could be done for a few representative catchments, and along the lines of the demonstrations in Clark and Kavetski (2010). If the editor thinks this is unnecessary detail, I would at least expect a more in-depth discussion about how the findings of Clark and Kavetski (2010) and Kavetski and Clark (2010) apply to the application in the manuscript.*
- ***Adaptive time stepping is difficult to realize in connection with AD, more specific in connection with parallel processing of minibatch optimization.*** *I never tested, but Chaopeng Chens explanation of this point makes sense to me. In my review, I never asked the authors to include such tests in the manuscript, therefore there is no disagreement here.*
- ***Discussion of HBV structural/functional changes (capillary rise) in the same paper.*** *In my review, I was mentioning that discussing structural/functional changes to the HBV model to solve an apparent model deficiency has little to do with the key message of the manuscript, and therefore suggested removing it. I still think that leaving this part away will help the paper to better convey its message. The reply by Chaopeng Shen - "Many article carry more than one stories and this is a beneficial (although not that major) improvements to the model. We do not want to write another article for this change." - has not convinced me otherwise. I will leave this decision to the editor.*

Yours sincerely, Uwe Ehret

We appreciate the comments from the reviewer. We still want to keep our paper as a research paper. We plan to add a more in-depth discussion to show the importance of the implicit scheme in large-scale hydrological simulation with big data and how our results align with the findings of Clark and Kavetski (2010) and Kavetski and Clark (2010), but we do not want to repeat their work that has been done thoroughly.

In this work, our focus is solely on daily hydrological modeling using conceptual models. The numerical schemes employed within the framework of the differentiable model need to maintain stability for simultaneous large-scale simulations in the minibatch while also allow for gradient tracking. We conducted additional tests on the differentiable HBV model with various numerical schemes and time steps (see appendix Table A2 and Figure A3). In theory, with smaller time steps, we were supposed to configure forcing functions to provide hourly inputs that match the progression of time within a day, i.e., the inputs should reflect diurnal changes in forcings. Even though hourly data are now publicly available, directly training an hourly model with ML techniques remains computationally expensive and may also cause the notorious problem of gradient vanishing (Gauch et al., 2021).

We included the following analyses in our discussion section and appendix:
1. *"We conducted tests on the differentiable HBV model with various numerical schemes and fixed smaller time steps (Table A2 & Figure A3 in Appendix). The sequential model and implicit adjoint model with a 1-day time step presented higher performance than the explicit Euler schemes with smaller time steps or the fourth-order Runge-Kutta scheme. The main reason may be that the daily forcing inputs and daily physical parameters from the neural network do not match the smaller time steps within a day. Thus, explicit schemes with smaller time steps may require matching forcing functions as well. Some multi-time-scale ML techniques have been used to predict hourly flood hydrographs using daily flow data to avoid gradient vanishing issues in the direct hourly training (Gauch et al., 2021; Sarıgöl and Katipoğlu, 2023). These approaches present possible solutions for future investigations." line{583-590}.* The implicit scheme yielded superior performance in terms of KGE and high and low flow metrics. Both reducing time steps and using implicit schemes improved model accuracy, aligning with the findings of Clark et al. in 2010.

2. An important finding in Kavetski and Clark (2010) is that numerical approximation errors can be compensated for by distorted parameter values during calibration, resulting in a 'right result for the wrong reasons.' This phenomenon can affect parameter uncertainty analysis, hinder meaningful parameter interpretation and regionalization, and lead to erroneous internal model dynamics. To explore this further, we examine the metric (KGE) surface within the 2D slice defined by field capacity (FC) and the parameter, derived from various numerical solutions (Figure A3). *"The parameterization function (the neural network) embedded in the differentiable models demonstrates robustness, as evidenced by the similarity of parameter patterns and metric surfaces derived from various numerical schemes in Figure 7 and Figure A3. We did not observe a notable macroscale roughness in the metric surface (Figure A3) as shown in Kavetski and Clark (2010) when using explicit schemes. Moderate distortions and roughness were present on the KGE surface in models employing the RK scheme (sites A and D). As we reduced the time steps and transitioned to implicit schemes, these distortions seem to have alleviated and converged toward the metric surface, consistent with the correct numerical solution. That is, the 4-hourly and hourly patterns are more similar to the implicit results than that of the RK scheme. The convergence toward the implicit scheme suggests that the implicit scheme results are more reliable." line{591-599}*

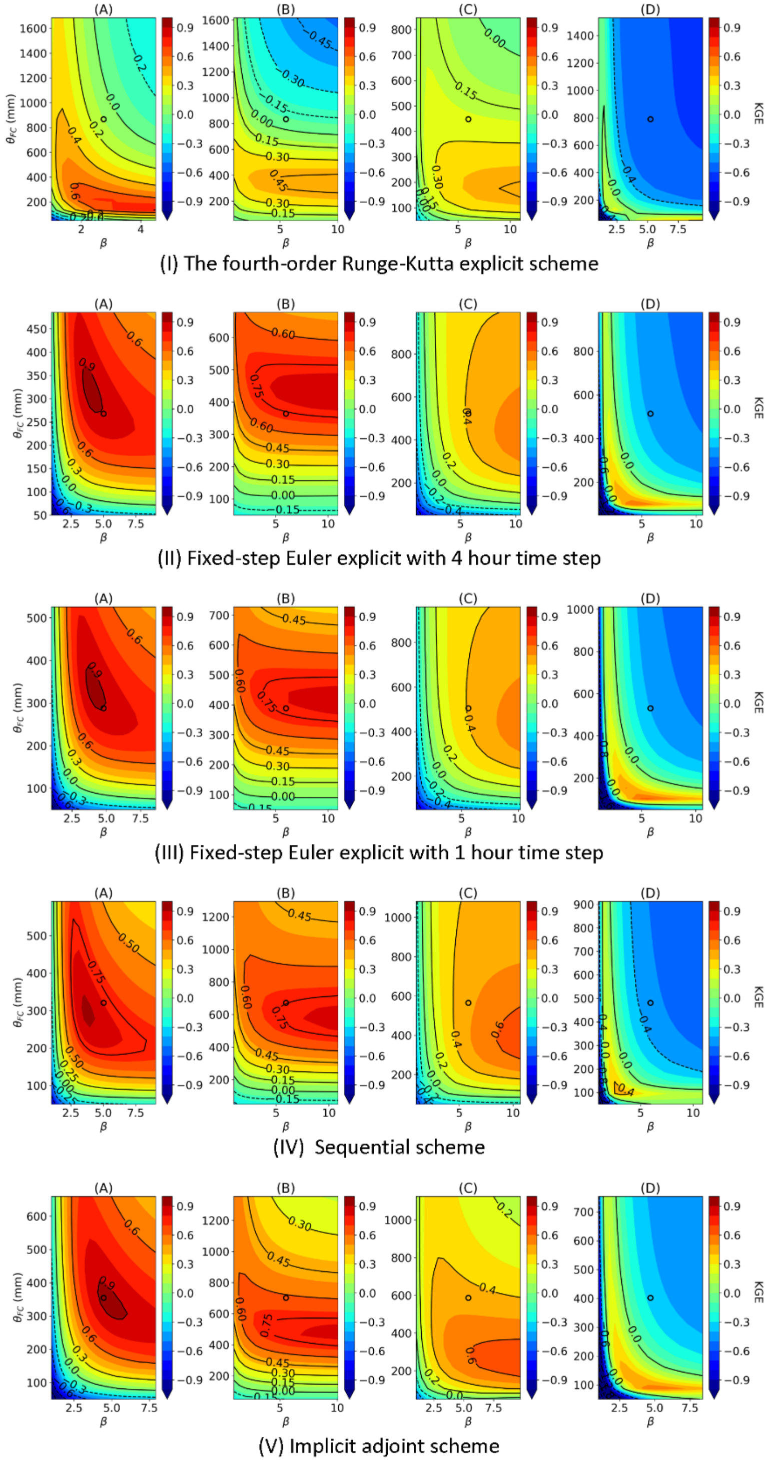
3. Even with small time steps, we still observe that explicit schemes exhibit parameter distortion and potentially problematic internal model dynamics, as demonstrated in Table 2 and Figure 7 in the manuscript. Such issues have implications on our ability to learn a better model structure, and hence we argue that the implicit scheme has value.

4. Regarding the reviewer's recommendation to drop the analysis about the structural change, we have some reservations. Because we are providing comprehensive comparisons both with the change and without the change, it does not seem like the readers would lose anything by

seeing the best configurations. In fact, it is to the benefit of the community to see what changes matter and by how much, and how to obtain the state of the art. While we fully respect the reviewer's opinion and could see where he is coming from, we respectfully would like to continue including this content. Nevertheless, in response to the reviewer' comments, we reduced the amount of discussion with respect to this component.

Table A2: Summary of streamflow metrics for models using different numerical schemes and time steps. Timing was obtained on a Nvidia Tesla V100 GPU.

δModel	Numerical scheme	Time step	Memory Usage per batch	Computational time per batch	Median NSE	Median KGE	Median low flow RMSE (mm/day)	Median peak flow RMSE (mm/day)
δHBV	Fixed-step explicit	1 day	2274M	1.6s	-	-	-	-
δHBV	The fourth-order Runge-Kutta explicit	1 day	2532M	3.9s	0.69	0.70	0.061	3.25
δHBV	Fixed-step explicit	4 hours	2706M	6.3s	0.72	0.71	0.09	2.50
δHBV	Fixed-step explicit	1 hours	4146M	18.1s	0.72	0.71	0.08	2.63
δHBV	Sequential	1 day	2266M	1.5s	0.73	0.73	0.074	2.56
δHBV.adj	Implicit adjoint	1 day	2788M	19.5s	0.72	0.75	0.048	2.47



(I) The fourth-order Runge-Kutta explicit scheme

(II) Fixed-step Euler explicit with 4 hour time step

(III) Fixed-step Euler explicit with 1 hour time step

(IV) Sequential scheme

(V) Implicit adjoint scheme

Figure A3: Impact of numerical schemes on the KGE surface of the HBV model: The contour of KGE calculated from the (I) 4th order Runge-Kutta explicit scheme, (II) Fixed-step Euler explicit with 4 hour time step with 4 hour time step, (III) Fixed-step Euler explicit with 1 hour time step, (IV) sequential scheme, and (V) implicit adjoint scheme on the 2D slice of field capacity (FC) and parameter. The predicted parameter values are positioned at the central point of the contours delineated by circles. The locations of selected sites are annotated in Figure 7.