Dear Referee #2,

Thank you for your reviewing and giving us so many useful advices. I appreciate your efforts. Now I put my response to your comments in following paragraphs.

Yonghe
2024-12-6

The comments of Referee #2

Review of NMP-Hydro 1.0: a C# language and Windows System based Ecohydrological Model Derived from Noah-MP
Recommendation: Major revisions
The authors have more-or-less replicated the Noah-MP model physics in a C# environment in an effort to expand the Noah-MP and WRF-Hydro research community to personal computers with Windows operating systems. This undertaking is generally reasonable, and a windows ready C# version of Noah-MP coupled with streamflow may be useful to hydro- researchers unfamiliar with Fortran and Linux based computing systems. Further, the results show overall "good enough" agreement between the legacy model and the replication to support/justify use of NMP-Hydro as a research tool.

Reply: Thank you for acknowledging our work.

However, there are significant unexplained differences between the two model frameworks that the authors disregard with minor speculation. The paper would be strengthened substantially if the authors actually tracked down the source of these differences and at least documented it as opposed to simply guessing that they are caused by "precision" differences. Such an effort could involve more isolated evaluation of the model components that seem to create these issues.

Reply: I understand your doubts. It is difficult to determine the actual source of some minor differences. As a model developer with over 16 years of experience, I have extensive knowledge of various types of programming code and know how to debug and identify errors or exceptions. If I can successfully identify the sources of these differences, I will not let them linger until the submission of the manuscript. My development of NMP-Hydro began in the January of 2018, and it has been six years now. During the past six years, I spent at least three months every year in discovering a large number of bugs and resolving code differences, which required me to put in a lot of hard work.

In the manuscript, we mentioned that we performed code validation using breakpoint debugging with the original WRF-Hydro model (although FORTRAN language does not have an efficient breakpoint debugging functionality, we can only use the print clause to print out the variable values). However, such debugging can only be completed in less than 3 time steps. In fact, after only one time step, "floating-point precision" causes small differences between the two models for some variable values, which poses great challenges.

As you suggested, 'isolated evaluation of the model components' is needed, but this suggestion is usually difficult to implement. In Noah-MP's code, there are indeed many functions divided, but these functions are nested with many complex functions. These functions are not pluggable modules, but rather have many relations between various variables across multiple functions. It is possible to perform error detection within the first running step by isolating different functions, but it is very difficult to perform after running hundreds or thousands of steps.

Fortunately, recently I conducted a new analysis by printing out certain identical variables in WRF-Hydro and our NMP-Hydro. It was only after running for tens months that I find the first significant inconsistency. I finally found a source of error: it was not in the code of the Noah-MP LSM, but at the entrance for calling Noah-MP LSM. Our previous code of NMP-Hydro does not support the real-time updates of the 'FICEOLD'. Now we can ensure that the 'FICEOLD' in our NMP-Hydro is updated in every running step, which eliminates many inconsistencies (Fig.1). Another source of error was identified: a mistranslation of the function 'SNOWH2O' regarding its parameters. The definition of this function in the original FORTRAN code contains several ambiguous points.

Unfortunately, a few inconsistencies remain. The specific source of these discrepancies is associated with the transition between the no-snow layer and the presence of a snow layer. The differences in the simulated output do not consistently manifest. For instance, beginning on January 1, 2000, the first significant error emerged in February 2001; however, when we initiated simulations from 2001 onward, this notable error did not occur.

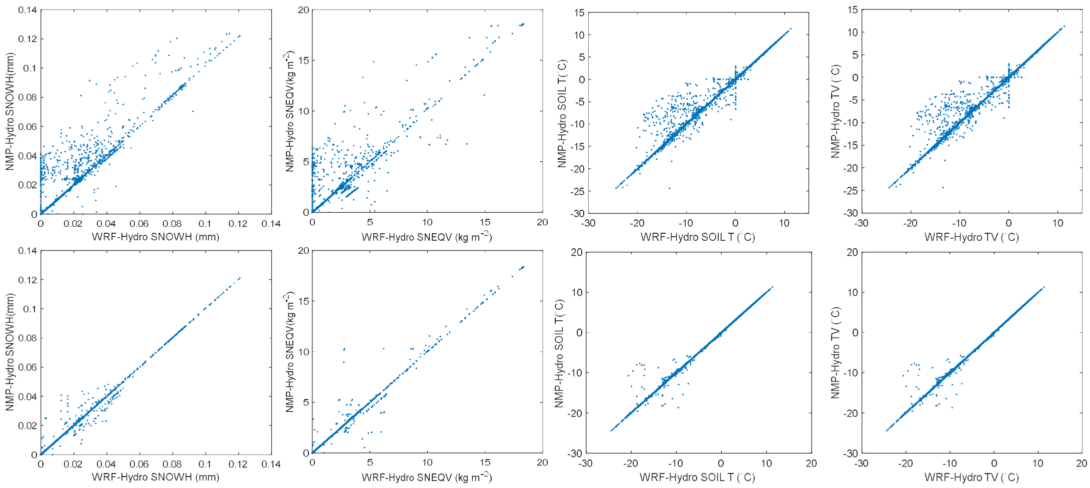

Fig.1 The comparison of variables simulated by WRF-Hydro and NMP-Hydro. The upper four: before the correction on the FICEOLD updating in NMP-Hydro; the lower four: after the correction on the FICEOLD updating.

Additionally, the paper would benefit from a cursory "speed" comparison between the WRF-Hydro version of Noah-MP and the NMP-Hydro version, such a comparison would help bolster the motivation for reproducing Noah-MP in C# beyond that of simply "some people don't like Fortran and Unix."

Reply: This comparison of the speed between the two models is also difficult to implement. I cannot use the two environments on the same computer, because WRF-Hydro usually runs in a Linux environment, while NMP-Hydro runs in a windows environment. Although my computer uses the windows 11 which supports Linux system (the WSL system), but I failed to correctly compile the code of WRF-Hydro on my WSL system.

I also do not think that users are very interested in comparison of running speed between the two models. Everyone knows that FORTRAN and C can run faster than C#, because

FORTRAN/C is a relatively low-level language compared to any modern object-oriented computer language. However, as a language that can run in native machine code, C# is not a slow one. The speed comparison of C# and Fortran (or C) can be found in many documents on the internet, therefore, there is no need to compare here. The difference in the two models' speed is mainly governed by the difference between the two languages. According to my experience (comparison between two different computers), WRF-Hydro indeed runs faster than NMP-Hydro when the latter runs in a non-parallel mode. NMP-Hydro can run in a parallel mode on a personal computer, while WRF-Hydro cannot. WRF-Hydro can run in a parallel mode using a MPI environment of high-performance computers, while NMP-Hydro does not support any MPI environment.

As for the speed comparison with different threads used, for NMP-Hydro, we have presented some tests in this revision:" *We tested the time it takes for NMP-Hydro to execute by setting multiple C# parallel threads. The computer used for the testing is a common laptop with 6 CPU cores. The results indicate that for the execution of the entire domain, as the number of threads increases from 1 to 6, the average time consumed for each time step is 1576ms, 977ms, 801ms, 711ms, 679ms, and 672ms, respectively. When the number of threads is set to 1, the time consumption is slightly greater than the time for the execution in the non- parallel mode (1461ms). It is worth noting that the time spent is not linearly related to the number of parallel threads, which can be explained by various reasons. One is that some tasks are actually not executed in parallel mode, such as reading meteorological input files. Another reason is that not all threads in NMP-Hydro are fully processed by the CPU core, as there are many other tasks in the entire Windows environment that have to be processed simultaneously by the same CPU cores."*

Overall, I recommend major revisions with a focus on identifying and discussing why certain model components do not behave exactly as they do in the Fortran environment, and on benchmarking model performance.

Reply: I also aim to diligently minimize any discrepancies between the two models prior to drafting the manuscript. However, I have already invested more 6 years in addressing these issues, and there are indeed some inconsistencies in the output. I find it quite challenging to completely resolve these problems based on my own strength. Furthermore, the Noah-MP model cannot be easily decomposed into multiple components as imagined, because many variables are intricately interconnected across multiple 'functions'. For instance, a minor variation in a specific variable can lead to significant inconsistencies after several time steps, making it difficult to identify the source of such discrepancies. From a technical perspective, capturing all subtle differences across different languages poses great challenges.

Fortunately, I have identified two major sources of the differences and resolved them now.

General comments:

I think the authors should strongly consider a different name for the tool than NMP-Hydro since this is extremely close to WRF-Hydro or NWM branding and is essentially a replica of WRF-Hydro system. This would help differentiate the two modeling systems and avoid confusion within the research community. Perhaps something that involves the C#, since that is the main novelty of the system presented here.

Reply: Considering that two reviewers, including you, believe that renaming is necessary, I now decide to rename it during the upload phase of this revision. Renaming the model is not an easy task, as the code has already been submitted to multiple sites. In this revision, we must use the original name.

In fact, I don't think the current name NMP-Hydro is very similar to WRF-Hydro or NWM. After all, many brand names nowadays are similar and there is no standard to avoid similarities. If we successfully change it to another name, will other reviewers or readers also have different opinions?

In the experimental configuration, the authors describe using a 6km grid for Noah-MP with a 1 degree meteorological forcing. Is there any meteorological downscaling performed within either WRF-Hydro or NMP-Hydro to reconcile these resolution differences If not, the simulations would effectively be running ~400 single-column Noah-MP runs with nearly identical meteorological inputs, and the only spatial-detail finer than 1 degree would come from differences in soil texture and land cover class. Further, some of the differences seen between the models, particularly in the winter season could be related to differences in downscaling, so I think it's important to at least clarify whether or not downscaling is applied.

Reply: The manuscript now describes this information in a clearer way. No downscaling was used here, and the 6-kilometer resolution of the driving dataset was only the result of regridding using bilinear interpolation. Whether to use downscaling is actually unrelated to the comparison. I just need to ensure that these two models use the same forced dataset (both downscaled and regridded data are acceptable).

It's not that surprising to me that the largest differences occur during the winter, though I suspect this has relatively little to do with the snow/frozen soil model physics, and rather may have to do with minor differences in the energy balance over snow that causes small differences in snow temperature to affect snow and surface albedo which can feedback into the energy balance and cause greater model divergence.

Reply: My analysis supports your viewpoint. In this revision, I found an important code inconsistency: the FICEOLD variable was not updated at each time step. The correction of FICEOLD update has eliminated a large number of inconsistencies, but there are still some inconsistencies. Although I cannot guarantee that these remainder inconsistencies are related to errors in the snow, based on code analysis, they are likely caused by the accumulation of inconsistencies in the snow (variables SNOWH and DZSNSO). However, it is difficult to find the reasons for the differences in snow variables, as many variables have small differences, and the growth of differences in snow variables seems to be moderate. It is highly probable that the differences in SNOWH is due to some local accumulation of floating-point errors.

In the code of Noah-MP, if SNOWH (the thickness of snow) is less than 0.025m, the ISNOW is zero (means no snow layer), otherwise, the ISNOW is 1 (means there is one snow layer). **Fig.**2 shows that the inconsistences in SNOWH cause different ISNOW (the layer number of top snow layer) shifts. The major inconsistences in snow equivalent (SNEQV), vegetation temperature (TV), the depth of the top snow layer (DZSNSO) are related to the inconsistences of ISNOW. This can explain that when there is a snow layer, it will give rise to different TV (TG (ground temperature) as well) and SNEQV values.

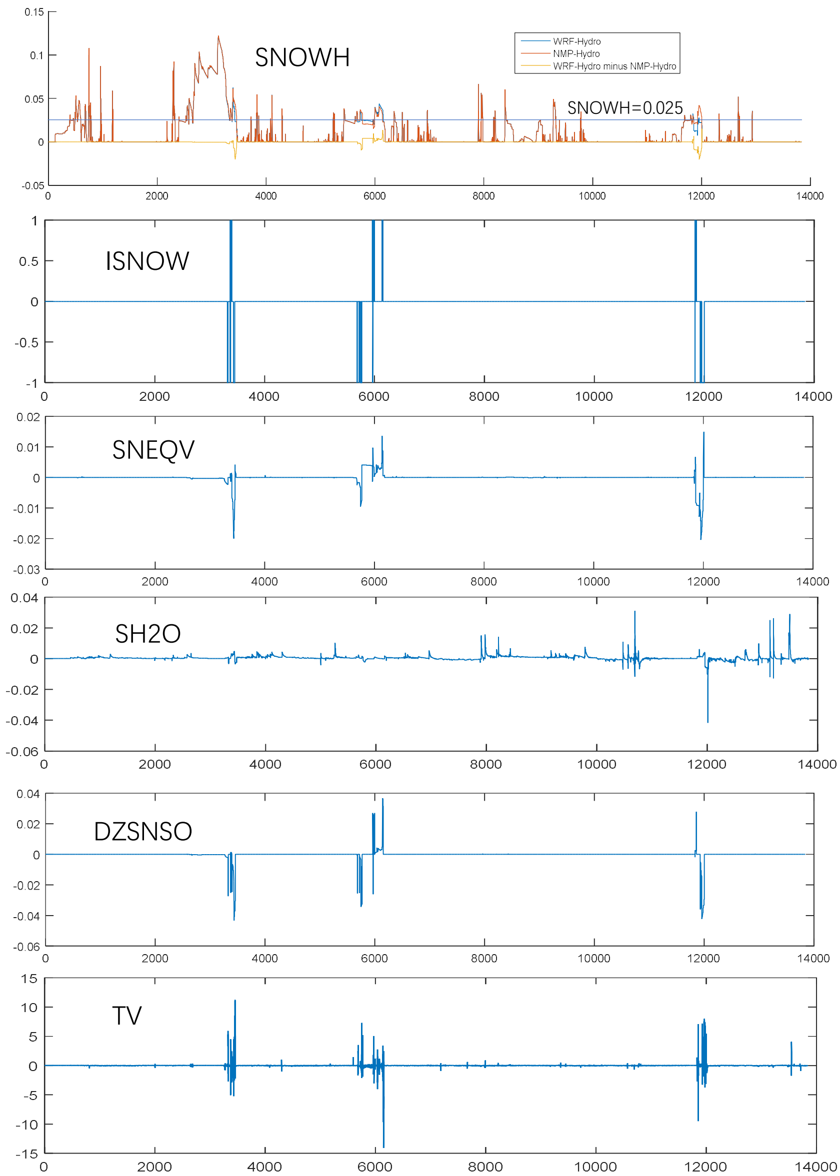

Fig.2 The differences (WRF-Hydro minus NMP-Hydro) between the variables simulated by NMP-Hydro and WRF-Hydro. The plots are based on three-hourly series, starting from 1 January, 2000.

Finally, is there any plan to maintain this version of Noah-MP to match new release versions (e.g., Noah-MP 5.0) as the Noah-MP developers at NCAR continue to expand the model capabilities. Even if the authors, justifiably, did not put effort into translating the new code structure associated with version 5.0, is there a plan to try and implement improved model physics into the NMP-Hydro framework. Some discussion around this topic may improve the manuscript.

Reply: Implementing new model physics into the model is necessary. However, I am the only developer for the development of NMP-Hydro and the task has spent more than 6 years. Testing new models by reference to a legacy model is a very time-consuming task and is not as easy as someone imagined. Therefore, I will not consider bigger plans in the future because it seems unrealistic for me now. In my opinion, there are always some uncertainties in any module of

physics, continuously adding more modules may not bring additional benefits to the scientific studies.
In this revision, I have added some plans which need to address in the future.

Specific Comments:
Lines 57 – 59: This sentence is somewhat misleading and mildly incorrect. True, NoahMP has been integrated seamlessly into the WRF model as a two-way coupled LSM to the atmospheric model and has been since its initial release ~2011/2012. However, WRF-hydro is related to WRF only insofar as it uses the same coding conventions and architecture and can be coupled into the WRF framework. WRF-Hydro itself is a system designed to couple atmospheric forcing to a distributed version of Noah-MP (or other LSMs) and routing/stream flow models. Consider rewording.
Reply: Thank you for pointing out the wrong description. I am not familiar with the deep details of these models. Now, I modified this description: *Based on Noah-MP, WRF-Hydro was developed, and can be seamlessly integrated into the Weather Research and Forecasting (WRF) model. (Gochis, 2020). Furthermore, the offline WRF-Hydro model plays a pivotal role in the National Water Model, contributing to the simulation of floods and river flows across the United States.*

Lines 68 – 70: additional wording concerns, specifically the words "coupled" and "models" in this context. HRLDAS and WRF-Hydro are not necessarily considered "models" so much they are frameworks used to couple various models together. Also, if you are going to mention the HRLDAS as a framework, consider also including the Land Information System (LIS) here. Consider rewording something like:
"Noah-MP is supported by several diGerent modeling architectures and frameworks to facilitate coupling it to various other Earth system modeling components including, WRF, MPAS, HRLDAS, LIS, and WRF-Hydro. This makes NoahMP a powerful research and forecasting tool within the hydrology community"
Reply: Based on your good advice. I modified the text accordingly.

Lines 70-72: This sentence is almost identical to a sentence on lines 59-60, please remove it.
Reply: Removed now.

Line 162: What is "each variable"? Soil temperature? Snow? Soil moisture? Surface Temperature? Energy balance?
Reply: Here the variables refer to all the variables in the programming code. Not only the physical variables as you mentioned, but many local variables that can influence the simulation. Now, the code is rewritten as "*Initially, the code underwent a meticulous step-by-step check by examining the printed values of many variables (including many local variables in the code) in WRF-Hydro 3.0 running for specific single columns.*"

Line 208: The word "comprising" should be "comprised"
Reply: Corrected.

Line 261: Please change "significant differences was" to "significant differences were"

Reply: Corrected.

Lines 288 – 289: What is meant by "disparate parameter configurations?" If I understand this correctly, does this mean that the parameter tables (i.e., MPTABLE.TBL, etc) that define specific snow/soil/vegetation properties might be different between WRF-Hydro and NMP-Hydro? This would be a huge issue trying to validate one against the other.

Reply: It refers to the hardcoded local parameters (usually local variables in a certain function). There are many of them. I have changed the sentence:" *Such discrepancies may be attributed to a number of factors, including floating-point calculation errors, some inconsistent hardcoded parameter values (as local variables in certain functions), or encoding inconsistencies.*"

Line 296 – 273: It seems unlikely to me as well that floating-point errors would result in large differences here, LSM's are tightly constrained by the forcing, unlike global atmospheric models for example, such that floating-point errors don't really cascade. The authors mention snow/frozen soil as potential reasons for the discrepancy, have you looked at snow/frozen soil variables related to runoff. For example, differences in snow melt, soil ice content or total soil moisture. The Noah-MP snow and frozen soil models are simple enough that I would not expect trouble when converting code over from one language to another.

Reply: Thank you for telling me that "LSM's are tightly constrained by the forcing". During this revision, I have made deeper trace on many variables. Although two major sources of difference were discerned and resolved now, but there are still differences left. These differences also are found to be related to the transition between no-snow layer and the presence of a snow layer. The difference in snow layer transition is caused by the differences in SNOWH, however, what causes the difference in SNOWH is difficult to resolve now. Generally, these differences do not cascade over many time steps.

Table 3: Please change "The first experiment" to "control"

Reply: Modified.

Lines 311-312: This line indicates that the authors compared NMP-Hydro with WRF-Hydro, but it's unclear to me that there is a model fun with WRF-Hydro in the experiment suite, rather it looks like a basic parameter-sensitivity study. Are the authors able to clarify which experiment is run with WRF-Hydro, or are all of the simulations presented in figures 9 and 10 NMP-Hydro? If that is the case, please edit this line to reflect that so there is no confusion.

Reply: This is a wrong description in the previous manuscript. No more comparison between the two models is presented in this section. The sentence is modified as "Here, we present the numerical outputs of NMP-Hydro on the streamflow discharges over the Yellow River, with various parameterization schemes were used."

Line 382: As I understand it here, the modeling isn't "Based on Noah-MP". It is Noah-MP, only recoded in the C# programming language and coupled to a streamflow model. Please edit to be clear about this point.

Reply: Your understanding is correct. We have made some revision on the descriptions over the entire manuscript.