Response to the comments of the reviewers

20.10.2022

Markus Köhli

Physikalisches Institut

Ruprecht-Karls-Universität Heidelberg

ANP-PAT

Martin Schön

Helmholtz Centre for Environmental Sciences, UFZ Leipzig

Dear Reviewer 2,

we are very grateful to the reviewer for the helpful comments and suggestions. In the following we address individually the comments to the manuscript 'URANOS v1.0 - the Ultra Rapid Adaptable Neutron-Only Simulation for Environmental Research' submitted to GMD. Reviewer's comments on the manuscript are bold, our answers italic and the latexdiff of the submitted paper indented in quotation.

Reviewer 2

This is a well developed manuscript, demonstrating the applicability of a novel though well-tested code. The manuscript is well readable, inspirational and easy to access for environmental scientists although containting dense information on physical processes and modelling. I recommend publication in GMD with minor revisions urging the authors to expand on a few minor points (see below) and add a new section on limitations and outlook. This section will benefit the readers and inspire the next generation of researchers to build on this code, increase the user space and the capabilities of CRNS to deliver accurate environmental observations.

Please address following questions (in the new section):
What is the run time on a standard architecture (e.g. intel Core i5, 8GB RAM, Windows or Unix)?
Is the code parallelized for HPC applications?
What are the most expensive calculations and how to they scale from 2D to 3D for a „simple" geometric set up?
Can the atmosphere and cosmic ray interactions be modelled using URANOS code?
What is needed and what is the uncertainty to expand the code to include uncertainty from the cosmic ray energy spectrum at the top of the atmosphere and further particles?
Given URANOS is applied to simulate each CRNS locations, what are the remaining major uncertainties constraining the accuracy of cosmic ray neutron sensor derived hydrogen content in the CRNS footprint?

This comment on the proposition of a new section is not clear to the authors. Most of the questions brought up by the reviewer are addressed in section 9 'Performance benchmarks'. The first question on the run time is answered in detail providing several scenarios, including the standardized simple setup with just air and water. URANOS can be compiled an run under Linux/OSX but we did not carry out specific performance evaluations.
The scaling for an HPC architecture is not addressed in detail. There are some general issues for effective parallelization of Monte Carlo calculations, that means achieving a performance gain beyond running several instances in parallel. A Monte Carlo tool like URANOS calculates the history of neutrons subsequently. Each step features highly computationally unequal branches. Moreover, a neutron can terminate after one or 1000 calculations. This makes it problematic to parallelize on the neutron or the interaction level. Furthermore, it is necessary to constantly switch cross section files and geometry for which none of them directly fits into the cache of the CPU. The memory of a GPU would be suitably large but the complex interaction calculations do

not go along the capabilities of typical shader units. There would be alternative possibilities of beneficial realizations on a GPU, however, the coding task for such would be a job on its own. Historically other methods have been used to increase the performance of Monte Carlo simulations like the MultiGroup method or the calculation in fixed energy groups. URANOS itself scales well with the number of physical cores involved. Every additional hyper-threaded core contributes with the equivalent of half to one full core. We have added one sentence with respect to this question.

The performance evaluation is likewise answered in the respective section regarding the model complexity. In general the usage of voxels mainly blows up the memory usage, not the computational performance. In case the model domain contains a manifold of different materials the performance decreases slightly but not significantly, which is one of the benefits of a voxel engine. One of the most expensive calculation is evaluation of the scattering angle as for each energy a function of two interpolated Lagrange functions constructed by a set of coefficients has to be evaluated and then randomly sampled. However, this only comes into play for the MeV region. Thermal scattering is also computationally demanding as for each neutron a magnitude more of function evaluations is necessary than in the epithermal case. Summed up, there is not a single process which can be pointed out, which would be a significant bottle neck. This fact, however, is result of months of performance optimizations, a process in which each possible lengthy calculation has been identified.
As URANOS can only model neutrons, it features an effective high-energy cascade model. The physcial description of neutrons for the atmosphere (as a gas) is physically correct. The effective high-energy model is based on PHITS and MCNP calculations and emulates the presence of other particles and their typical neutron-generating effects. The mentioned other commonly used codes, however, do also not feature physical calculations of all particles and their respective processes. They break down typical possible interactions into analytical models and fine tune them to measured data. Moreover, many cross sections, especially of elements of high abundance in the environment, are missing, incorrect or feature high uncertainties. Therefore each code has its own methods to overcome those shortcomings. As to how much these models are appropriate representations is subject to ongoing studies. In order to provide at least a lower limit for uncertainties, we expand the example given in the manuscript: the best known cross section is the one associated with the elastic scattering of hydrogen, around 0.3 % statistical uncertainty. Other isotopes are in the range of 1 % or more. Inelastic scattering cross sections can easily reach 10 %. Considering such variations on the cross sections, typical quantities derived from a simulation has a relative error in the range of 1-3 %. Realistically speaking the absolute error of this kind of simulations can be estimated to lie in the order of 5-10 %. As, specifically speaking, CRNS measures relative differences the error on that lies significantly below that and is currently at least lower than the error on any other quantity used for comparisons between simulation and experiment, especially the soil moisture distribution horizontally and vertically within the footprint.

Line 132: It is unclear for the reader, how TRandom3 is programmed, language, and what a "modern architecture" is. Are the authors referring to HPC systems, GPU based HPCs or Laptops build in 2019? Please specify what you refer to as modern architecture ideally in the flops as measure for computational performance.

In order to provide you some 'extreme' examples: On a workstation CPU, Xeon E5-1620v2 from 2013 with 3.7 GHz scaling to 3.9 GHz, each call takes around 14 ns. On a Laptop CPU from 2020, Intel Core i5-1135G7 with 2.4 GHz scaling to 4.2 GHz (typically not reached), each call takes around 11 ns. This example shows that flops, even broken down to a single core operation, is not a good measure to characterize the performance here. The reasons for this behavior is that CPUs

are not simply 'number crunchers' and going into detail here on to what exactly such differences arise would go beyond the scope. The different call times shown in this example above will not influence the performance of URANOS significantly. Therefore we used the term 'modern architecture' to overcome that problem while still showing its performance. However, we additionally have provided a time range for the 'architecture' (CPU (< 5 years)).
In case the reader wants to know how TRandom3 is programmed and in which language, we have provided a reference to its implementation the framework ROOT, which is open source and therefore the code for TRandom3 can easily be accessed. In case the reader is interested in architectural considerations of the algorithm itself we provided a reference to the description of the Mersenne Twister.

Line 133: Please also mention possible other random number generators and are they available? If those are not relevant questions, then it seems the technical details are not needed and I recommend to simply state that „the TRandom3 random number generator is used" and remove the technical details on random number generation from the manuscript in this paragraph 2.2.

The random generator is the heart of a Monte Carlo code. Its working principles may not appear obvious to the user, but its technical details are relevant for experts. Although we agree with the reviewer that a detailed description of the use of random generators in Monte Carlo codes would go beyond the focus of the manuscript, we consider at least the specification of such basic features as speed and period length to be relevant. We want to show to the reader, that the performance is suitable for this application, with 10 ns per call other calculations tend to be the dominating factor. With the given period length and easily 1000 calls per neutron its randomness is assured for typical runs of 1e9 neutrons. By stating that it is seeded with the system time in ms, we indicate that by running several instances of URANOS started from the command line having the same initiation time, the randomness is not assured. ROOT features implementations of 9 different Random generators. Changing the random generator in URANOS is possible, one declaration at the program initiation needs to be changed, however, URANOS needs to be recompiled then.

Line 208: Please clarify what „MT numbers" are. Random „Mersenne Twister" numbers would not define reaction types, I assume.

In nuclear science both identifiers are common, in environmental science not. Sometimes in such interdisciplinary works, one faces the difficulty in standing at the interface between two different languages. Yet, we think that if the sentence „The ENDF format uses MT numbers to define reaction types and MF numbers to classify the data type of the respective set" is not descriptive enough the reference to the ENDF formats manual provided here is suitable to describe the MT and MF, material type and material file, numbers. In order to better guide the reader we have put both in apostrophes.

> The ENDF format uses the 'MT numbers' to identify neutron reaction types and 'MF numbers' to classify the data type of the respective file set.

Line 215: What is ensemble statistics?

We have added:

> Such require a very large number of particles to derive laws without necessarily taking into account each individual state vector.

Line 219: Please state which relevant and non-relevant interactions you are referring and what are the „two different types“?

We have added:

(among them absorption, elastic and inelastic scattering as well as evaporation)

Line 222: Please clarify, how is it possible that myons are not contributing while myons are the major cosmic rays entering the atmosphere, and neutrons are only a product of myon inter-action?

Thank you for that question. High-Energy (primary) neutrons are generated through several interaction channels. Evaporation neutrons are, however, mainly generated by other neutrons, protons and myons. The production probability decreases in the same order as the particle species is mentioned with myons contributing a few percent. Myons are the most abundant particles due to their low interaction probability (long mean free path). With a low interaction probability, the probability of producing neutrons is likewise low. In total one can describe it in the following way: the high energy cascade contributes to the production of neutrons by three different particle species with three different fractions and attenuation lengths (around 140 cm^2/g for neutrons, around 110 cm^2/g for protons and around 500 cm^2/g for myons). Future work will address these different attenuation lengths and model the high-energy cascade accordingly.

Line 332: Please state recommended default starting angle.

There might be a confusion here. The source options listed in the respective chapter are 'artificial' sources not related to the cosmic neutron source. For artificial source definitions which are relevant for for example detector simulations you want to tailor the neutron beam accordingly. The cosmic-ray neutron source, which is discussed in the following section, is based on analytical functions. Here, the source angle cannot be specified as it is intrinsically set by the description by Sato.

Thank you very much for the review of our manuscript.