Geoscientific Model Development Discussions

Geosci. Model Dev. Discuss., referee comment RC1
https://doi.org/10.5194/gmd-2021-162-RC1, 2021
© Author(s) 2021. This work is distributed under
the Creative Commons Attribution 4.0 License.

Comment on gmd-2021-162

Anonymous Referee #1

Referee comment on "POET (v0.1): speedup of many-core parallel reactive transport simulations with fast DHT lookups" by Marco De Lucia et al., Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2021-162-RC1, 2021

This paper targeted at a computing-demanding application, the geochemical part of the coupled reactive transport simulations, and developed POET (POtsdam rEactive Transport), a research parallel reactive transport simulator integrating algorithmic improvements. Due to several innovations proposed in this work, especially the methods to resolve issues from the overall load balance, the performance of the coupled simulations was improved, so the simulation time is reduced.

Major technical contributions and strengths of the paper include,

Selected application, chemical part of a coupled reactive transport simulation, is a challenging hotspot that demanding novel HPC solutions.

A master/worker design and a load distribution algorithm are proposed to well address the major challenges, ensure computational efficiency on both multicore and cluster compute environments, and achieve speedups of the selected application.

A fast MPI-based DHT, and the concept of approximated lookups, are proposed to enable caching efficiency and data reuse.

The performance of using all above innovations is able to be improved, while large-scale simulation can be potentially more beneficial.

In addition, the paper is well-organized, and well-written, with sufficient figures and tables

to support descriptions, enough material to validate the results, and sufficient related work to prove the state-of-the-art. As a computer scientists, I would say that proposed techniques have make the best of the hardware architecture and resources, as well as the algorithmic features, so the existing challenges and bottlenecks are well addressed. The work can be a good guidance for optimizing corresponding models using HPC systems based on CPUs.

One shortage is that the largest run only occupies 719 core. The parallel size is too small to demonstrate the benefits this work can actually obtain in terms of large-scale simulation. My other concern is the portability of this work. Can proposed methods be applicable to other many-core-based systems, such as CPU-GPU heterogeneous system?