Reply to Referee 2:

Thank you for your comments on the manuscript. The questions and suggestions you raised are very detailed and provide practical guidance.

Referee 2: This manuscript summarizes an exercise in porting the WRF-CAMx modeling system to a specific computational platform, namely Loongson 3A4000 CPU platform with MIPS64 architecture. Model simulations of evolution of air pollution over a period of 72-hour duration over a domain encompassing the Beijing-Tianjin-Hebei region are conducted on this platform and benchmarked against comparable simulations on a X86 platform. Additional simulations for a much shorter 2h period with CAMx are also conducted to examine parallel performance on the different architectures. The authors present a variety of standard statistical measures to demonstrate the cross-platform porting of the WRF-CAMx to the Longsoon 3A4000 platform they use. The manuscript is generally well written and clearly describes the work conducted by the authors. However, in my assessment the manuscript lacks scientific novelty and does little to advance either the development and evaluation of the modeling systems examined or in providing a robust assessment of the execution of these models on emerging architectures. In my view, much of what is presented, is a standard exercise in porting a numerical modeling system to a different computational platform and steps that are routinely undertaken to establish benchmarking on such systems – these tests are commonplace for both the WRF and the CAMx models used here as well as other air pollution modeling systems and are routinely conducted by their respective user communities. While the successful porting of these specific models (WRF v4.0 and CAMxv6.10) to a specific platform (Loongson 3A4000 CPU) may likely be of interest to a segment of the users of these models who may also be planning to use these specific MIPS CPU platforms, such assessments are commonly discussed in the user forums of these (and similar) modeling systems – I thus struggle to identify the scientific and technical uniqueness of this contribution.

Reply: The main innovation of this article is the implementation of a relatively comprehensive, moderately scaled, and highly practical scientific computing application based on the MIPS platform, also a new architecture, named LoongArch and mentioned by another referee, will be added in the revised manuscript.

This work enhances the adaptability of the modeling system to emerging platforms, making the models more robust. RISC, as an emerging CPU architecture, holds vast potential in high-performance computing. However, current high-performance computing platforms are predominantly led by the X86 architecture. Commonly used models like WRF and CAMx have primarily been developed for X86-based platforms, lacking adaptation for emerging architectures. Therefore, the adaptation to specific RISC platform was approached comprehensively, considering architecture, compiler, compilation options, and model code. MIPS is one of the earliest commercial RISC architectures, and the Loongson 3A4000 CPU, representing MIPS processors in recent years, has been selected for this study. Through literature review, it was found that there is limited research on the application of scientific computing on MIPS platforms, with only a few studies focusing on the FFT algorithm. Thus our study has constructed a numerical modeling system with WRF-CAMx modelling system, which encompasses the runtime environment, models, control scripts, and environmental data. It can be practically utilized for operational numerical simulation, representing a relatively comprehensive scientific computing application. This study could provide methodological and technological references for scientific computing applications on MIPS and other RISC platforms, such as RISC-V and LoongArch.

However, the hardware performance limitation of the MIPS platform and the immaturity of its ecosystem are factors that cannot be ignored when assessing the modeling system on emerging architectures.

Referee 2: No new developments to either the WRF or CAMx models are described, neither were any changes implemented to the respective model codes to improve their computational performance on the architectures examined. Rather changes were made to configuration/makefiles to facilitate the compiling of the model codes, which is somewhat standard practice whenever a model code is ported across platforms or when compilers are updated.

Reply: To build the WRF-CAMx modeling system on the MIPS platform, the developments in the WRF and CAMx model primarily include:

(1) Establish the runtime environment on emerging platforms, including parallel computing libraries such as MPICH3 (v3.4) and data format libraries such as HDF5 (v1.15.1) and NETCDF (C-v4.8.1, Fortran-v4.5.3). These libraries do not support the architecture (mips64el and LoongArch) and GNU compiler of Loongson platform. Relevant information needs to be added to the free software config.guess and config.sub provided by GNU org (part of the information is shown in Figure 1 for example, which can help identify the platform architecture and system during the compilation and installation of libraries using Configure and Make tools.

```
    ;;
loongarch32:Linux:*:* | loongarch64:Linux:*:*)
    GUESS=$UNAME_MACHINE-unknown-linux-$LIBC
    ;;
    ..
mips64el:Linux:*:*)
    GUESS=$UNAME_MACHINE-unknown-linux-$LIBC
    ;;
```

Figure 1. Information about the architecture of Loongson platform

(2) Incorporate architecture-specific settings for the model. Taking the meteorological model WRF for example, the architecture presets are stored in the configure.defaults file, but seetings about the Loongson platform is

not included. Specific architecture details, including GNU compiler and compilation options, need to be added, which can ensure the correct display of configuration during building WRF model, and part of information is shown in Figure 2 and Figure 3.

```
#ARCH    Linux mips64, gfortran compiler with gcc  #serial smpar dmpar dm+sm
#
DESCRIPTION     =       GNU ($SFC/$SCC)
DMPARALLEL      =       # 1
OMPCPP          =       # -D_OPENMP
OMP             =       # -fopenmp
OMPCC           =       # -fopenmp
SFC             =       gfortran
SCC             =       gcc
CCOMP           =       gcc
DM_FC           =       mpif90 -f90=$(SFC)
DM_CC           =       mpicc -cc=$(SCC)
FC              =       CONFIGURE_FC
CC              =       CONFIGURE_CC
LD              =       $(FC)
RWORDSIZE       =       CONFIGURE_RWORDSIZE
PROMOTION       =       #-fdefault-real-8
ARCH_LOCAL      =       -DNONSTANDARD_SYSTEM_SUBR  -DWRF_USE_CLM
CFLAGS_LOCAL    =       -w -O3 -c
LDFLAGS_LOCAL   =
CPLUSPLUSLIB    =
ESMF_LDFLAG     =       $(CPLUSPLUSLIB)
FCOPTIM         =       -O2 -ftree-vectorize -funroll-loops
FCREDUCEDOPT    =       $(FCOPTIM)
FCNOOPT         =       -O0
```

Figure 2. Presets for the Loongson3A4000 platform.

```
------------------------------------------------------------------------
Please select from among the following Linux mips64 options:

  1. (serial)   2. (smpar)   3. (dmpar)   4. (dm+sm)   GNU (gfortran/gcc)

Enter selection [1-4] :
```

Figure 3. The display of WRF configuration on Loongson3A4000 platform.

(3) Modify the code to make it run smoothly on a specific platform. Taking some function in the CAMx model for example, the model frequently uses

the “write” function for formatted output. The format specifiers in the parameters consist of data types (I, F, E, A, X, etc.) followed by a character width. In the CAMx model, the format specifiers in the write function mostly default to character width, but there is a compilation issue with MIPS GNU, requiring character width descriptors. It is also essential to ensure consistency with the default precision. A specific example is illustrated in the figure below. A specific example is showed in Figure 4.

Before modification:

```
write (iout,'(a,2a)') ' spec','total [ug/m3]','c* [ug/m3]   '
write (iout,'(i5,2e)') (idx(i),sctot(i),scsat(i),i=1,nsol)
write (iout,'(a,2e)') ' cpre,cpx ',cpre,cpx
```

After modification:

```
write (iout,'(a5,2a15)') ' spec','total [ug/m3]','c* [ug/m3]   '
write (iout,'(i5,2e15.7)') (idx(i),sctot(i),scsat(i),i=1,nsol)
write (iout,'(a5,2e15.7)') ' cpre,cpx ',cpre,cpx
```

Figure 4. The modification of format specifiers in “write” function

However, the description of the modification and development of the model is not detailed enough in the manuscript, and some technical details will be added in the attachment or revised manuscript. Furthermore, due to the hardware and ecosystem limitations of the MIPS platform, the improvement of model performance primarily relies on specific optimization options provided by the compiler based on the MIPS instruction set, which are mentioned in the compiler description of the manuscript. In the future, there will be considerations to modify the code for optimization.

Referee 2: It could be argued that running and porting of models across platforms and establishing the "reproducibility" of results through the benchmarking described falls under the scope of "development and technical papers", but there too the simulation durations and domain coverage are somewhat limited to clearly assess all technical aspects of running the models on the new architecture.

Reply: Due to the limited performance of the Loongson 3A4000 CPU at present, the simulation duration and domain coverage are somewhat restricted. In this study, we set up a 72-hour simulation case covering the Beijing-Tianjin-Hebei region. The chosen computational scale and simulation duration are moderate for the platform, facilitating the comparative analysis and performance evaluation. This case serves as a representative example of short-term regional numerical simulation application, which is sufficient for validating the feasibility of numerical simulation on the MIPS platform. The cases of broader coverage and longer duration will be considered to test on platforms with more CPUs.

Referee 2: At several places in the manuscript discussion, it is mentioned that MIPS architectures and the Loongson 3A4000 have the advantage of energy efficiency. However, the simulation design (domain size and simulation duration) does not appear to lend itself to adequately assess potential energy savings. Neither is any analyses presented to robustly infer the tradeoff between computational performance (since that seems to be poorer for the MIPS system used here relative to the X86 platform) and energy savings that may result from transitioning to such a platform.

Reply: Currently, the MIPS platform lacks effective tools for energy efficiency assessment. Rough estimates can only be made through the CPU's TDP parameters and performance evaluation of specific case. However, compared to CISC architecture, the high energy efficiency of RISC is determined by the characteristics of instruction set. Therefore, MIPS has a natural energy efficiency advantage about the architecture compared to X86. In the future, we will try to find a MIPS platform with a similar number of CPU cores as the X86 platform to comprehensively evaluate potential energy savings.

L113-115: This statement implies that the WRF-CAMx modeling system was developed in Xi'an, China and Milan, Europe – is that an accurate representation of the origin of these models or their linkage? Did Ramboll not develop the requisite code to link CAMx with WRF output?

Reply: The WRF-CAMx modeling system mentioned in this manuscript refers to an operational system of numerical forecasting designed for specific region, not just a simple linkage of the WRF and CAMx models. There are also a series of tools which support the operation of the modeling system. Their functions mainly include: Firstly, the configuration of models, which primarily involves the unified settings for the simulation domains and grids for Beijing-Tianjin-Hebei region, and the parameterization schemes suitable for this region. Secondly, emission data processing. The tool of SMOKE is used to process the emission source data to the simulation grids. Additionally, the C Shell scripts are used for system control and modules interconnecting in the modeling system. For example, in our 'run_CAMx.csh' script, it will check whether the required data is prepared; if not, it returns to execute the pre-processing modules. The link program provided by Ramboll serves as a part of these tools. This statement may not be clear enough. Here, the references of the WRF-CAMx modeling system used in Xi'an, China, and Milan, Europe, indicate that our system presented in the manuscript holds significant application value.

L303: “stability and availability” should be clearly defined. Is a single 72-hour simulation duration sufficiently long to test the stability of a model on an architecture?

Reply: Since our platform is equipped with only one CPU, its computational capacity is limited. The simulation cases are configured with relatively short durations, providing a moderate computation scale for this platform. And a simulation duration of 72 hours is already sufficient to the short-term numerical forecasting, and also for testing the stability of the model system. This is recognized in the research and application of numerical models. In the research of Wang et al. in 2019, a 72-hour case was set for the scientific validation and performance evaluation of the chemical transport module.

L456-457: How does the parallel performance of CAMx vary with problem size, i.e., number of grid cells? What fraction of the time is spent in output operations? Is it possible that with increasing computational size, a single processor would require more time than the configuration with two with one dedicated to I/O? How generalizable are the findings on parallel performance based on the limited domain size and simulation period?

Reply: In fact, the number of grid cells does objectively impact the model's parallelism, and the computational scale is one of the factors influencing the parallel performance of the model on a specific platform. However, due to our platform being equipped with one quad-core processor, and the limited computational scale of the test case, we have not encountered performance issues related to factors such as computational scale, I/O, and multiprocessors. Your considerations are comprehensive, and the questions are very meaningful, guiding us for further research in these directions.

L468-474: It is interesting that the performance of the MIPS platform decreased significantly when the number of parallel processes exceed 4. Is this unique to the Loongson 3A4000 or is this generalizable to other MIPS systems? Would the same hold for a domain with significantly larger number of grid cells?

Reply: The reason for this phenomenon is that the Loongson 3A4000 CPU has only 4 cores, when running with four parallel processes, each core's usage approaches 100%. When the number of parallel processes exceeds 4, processes will compete for computing resources, resulting in additional overhead and performance degradation. If we are interested in this, we need more MIPS or LoongArch CPUs, but the prerequisite is to make the sponsor feel that it is a valuable thing. For other MIPS platforms, it depends on testing results.

Citation:

Wang, H., Lin, J., Wu, Q., Chen, H., Tang, X., Wang, Z., Chen, X., Cheng, H., and Wang, L.: MP CBM-Z V1.0: design for a new Carbon Bond Mechanism Z (CBM-Z) gas-phase chemical mechanism architecture for next-generation processors, Geoscientific Model Development, 12, 749–764, https://doi.org/10.5194/gmd-12-749-2019, 2019.