Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

Geoscientific Model Development Discussions

MP CBM-Z V1.0: design for a new CBM-Z gas-phase chemical mechanism architecture for next generation processors

Hui Wang[1], Junmin Lin[2], Qizhong Wu[1], Huansheng Chen[3], Xiao Tang[3], Zifa Wang[3], Xueshun Chen[3], Huaqiong Cheng[1], Lanning Wang[1]

[1]College of Global Change and Earth System Science, Joint Center for Global Changes Studies, Beijing Normal University, Beijing 100875, China
[2]Intel (China) Corporation, Beijing 100013, China
[3]State Key Laboratory of Atmospheric Boundary Layer Physics and Atmospheric Chemistry, Institute of Atmospheric Physics, Chinese Academy of Sciences, Beijing 100029, China

Correspondence to: Qizhong Wu (wqizhong@bnu.edu.cn) & Huansheng Chen(chenhuansheng@mail.iap.ac.cn)

Abstract. Precise and rapid air quality simulation and forecasting are limited by the computation performance of the air quality model, and the gas-phase chemistry module is the most time-consuming function in the air quality model. In this study, we designed a new framework for the widely used Carbon Bond Mechanism Z (CBM-Z) gas-phase chemical kinetics kernel to adapt the Single Instruction Multiple Data (SIMD) technology in the next-generation processors for improving its calculation performance. The optimization implements the fine-grain level parallelization of CBM-Z by improving its vectorization ability. Through constructing loops and integrating the main branches, e.g. diverse chemistry sub-schemes, multiple spatial points in the model can be operated simultaneously on vector processing units (VPU). The Intel Xeon E5-2697 V4 CPU and Intel Xeon Phi 7250 Knight Landing (KNL) are used as the benchmark processors. The validation of the model outputs indicates that the relative errors are in an acceptable range (<0.05%). The results show that the optimization resulted in a 4.24x speedup on a single CPU core and 17.33x speedup on a single KNL core. For the node, the speedup on the CPU can reach 113.42x using Message Passing Interface (MPI) and 118.13x using OpenMP, and the speedup on the KNL node can reach 170.31x using MPI and 179.95x using OpenMP. The speedup of the optimized CBM-Z is approximately 50~52% higher on a 1-socket KNL platform than on a 2-socket CPU platform. This work improves the performance of the CBM-Z chemical kinetics kernel as well as the calculation efficiency of the air quality model, which can directly improve the practical value of the air quality model in scientific simulation and routine forecasting. Furthermore, since this optimization seeks to improve the utilization of the VPU, the model is more suitable for the new generation processors adopting the more advanced SIMD technology.

1 Introduction

Air pollution and its impacts on human health have attracted widespread attention all over the world, especially in developing countries (Gurjar et al., 2016;Zhang et al., 2017). As a useful tool for air quality problems, the Chemical Transport Model (CTM) is widely used in studies of air quality (Gao et al., 2016;Chen et al., 2015;Wu et al., 2014) and in establishing air quality

forecasting (AQF) systems. As the core of the AQF system, the CTM requires a large number of computation resources to simulate the complex chemical and physical processes. To satisfy the demand of routine air quality forecasting in a timely manner, coarse spatial resolution and relative simple processes in the CTM are adopted to minimize the use of computation resources. Meanwhile, the corresponding simulation studies with complicated processes are also limited by computation resources. Therefore, air quality simulation studies can benefit significantly from a calculation performance improvement of the CTM.

In the CTM, the most time-consuming module is the gas-phase chemistry module (Wang et al., 2017). The gas-phase chemistry module is described as a system of ordinary differential equations (ODEs) to simulate the chemical kinetics of trace gases in an atmosphere model (Seinfeld and Pandis, 2012). Linford (2013) reported that the Regional Acid Deposition Model version 2 (RADM2) (Zimmermann and Poppe, 1994;Chang et al., 1987), a chemical kinetics kernel, accounted for 90% of the computational time in the Weather Forecasting and Research/Chemistry (WRF-chem) model (Grell et al., 2005). Another widely used chemical kinetics kernel, Carbon Bond Mechanism version Z (CBM-Z) (Zaveri and Peters, 1999), accounts for approximately 68% of the computation time in the Global Nested Air Quality Prediction Model System (GNAQPMS) model(Chen et al., 2015;Wang et al., 2017). Therefore, accelerating the gas-phase chemistry module can directly improve the performance of the CTM as well as the whole AQF system. The AQF system can also benefit from the performance improvement by adopting a higher model resolution, thereby improving the frequency of air quality forecasting.

The performance of models improves with updated hardware. However, reaching the bottleneck of power density and the thermal limitation of the silicon technology for a single-core design, frequent updating has not been an efficient way to improve the scientific model's performance. Additionally, multi-core architecture and a heterogeneous computing architecture such as a Many Integrated Core (MIC) and a Graphic Processing Unit (GPU) have become the hardware trend for high-performance computing. Meanwhile, to take full advantage of the advanced features of new processor architecture, the applications or the models must be redesigned or rewritten. Xu et al. (2015) rewrote the Princeton Ocean Model (POM) using CUDA-C to port it from a CPU to a GPU platform. Linford (2013) also tried to solve the computation bottleneck of RADM2 mentioned above by using a heterogeneous platform such as GPU-CPU. In addition, our previous work showed the primary optimizations we performed to accelerate the GNAQPMS model on the new generation CPU and Intel MIC platforms (Knights Landing, KNL(Sodani et al., 2016)), and had a significant performance improvement on both platforms, a 2.77x speedup on CPU, and a 3.51x speedup on the KNL node (Wang et al., 2017). In this study, we redesign the code structure of the chemical kinetics kernel CBM-Z to improve its vectorization performance on the CPU and KNL platforms, which significantly improves its performance by fully utilizing the Single Instruction Multiple Data (SIMD) technology. We extracted this optimized CBM-Z module from GNAQPMS and made an independent model to test its performance, and the codes only contained this single module make it easier to let the air quality model developers to reuse the codes.

Section 2.1 in this paper introduces the detail of CBM-Z scheme adopted in GNAQPMS, and section 2.2 describes the new architecture we designed for CBM-Z. Since multiple spatial points were operated simultaneous in the optimized CBM-Z scheme, the optimized CBM-Z schemes were called Multiple-Points CBM-Z Version 1.0 (MP CBM-Z V1.0). In section 3.1

and 3.2, we present our benchmark platforms and the test case, and the test results are shown in section 3.3. The conclusions and discussions are given in section 4.

2 Method Description

CBM-Z is a core module in the air quality model that simulates the complicated gas-phase chemical processes in the atmosphere. In this module, too many branches and unbalanced calculations within the model grids make it a challenge to improve its performance on a vectorization level. This leads to the low performance of CBM-Z on the new generation processors that are highly dependent on powerful vector processing units (VPU). In our previous work, we conducted several optimizations on CBM-Z to enhance its vectorization and parallel performance(Wang et al., 2017). In this work, we attempt to further enhance its vector calculation ability by constructing a new architecture, which makes the CBM-Z module suitable to be vectorized. The CBM-Z module was extracted as an individual box model to test its performance and improve code reusability.

2.1 Description of CBM-Z

CBM-Z is a lumped-structure photochemical mechanism that was developed to meet the needs of city-scale to global-scale tropospherir chemical simulation (Zaveri and Peters, 1999). The original scheme contains 67 species and 132 reactions. CBM-Z has been widely used in the CTM, e.g., the WRF-Chem (San José et al., 2015), the Nested Air Quality Prediction Model System (NAQPMS) (Wang et al., 2001) and the GNAQPMS model. In the NAQPMS and GNAQPMS model, CBM-Z was further modified by Li et al. (2012). It was updated to 76 species, and 28 heterogeneous reactions were added. The CBM-Z solver uses the Modified Backward Euler (MBE) solver developed by Feng et al. (2015), a faster and more robust algorithm which overcomes inflexibility and preserves the non-negativity.

The main control flow of CBM-Z is shown in Figure 1. The ***IntegrateChemistry*** function is treated as the core-function of the module. CBM-Z contains five chemistry sub-schemes. They are the Common Chemistry Scheme (COM), the Urban Chemistry Scheme (URB), the Biogenic Chemistry Scheme (BIO), the Marine Chemistry Scheme (MAR) and the Heterogeneous Chemistry Scheme (HET). The integration of different sub-schemes is used to satisfy the simulation of diverse scenarios and scales. The combination of sub-schemes relies on the concentration and emission of each chemical species in the specific model grid, which is implemented in the ***SelectGasRegime*** function. The variable ***iregime*** stores the return-value of ***SelectGasRegime*** and controls the subsequent calculation processes of CBM-Z. The possible values and the sub-schemes represented are shown in Table 1. The combinations include the COM and HET schemes, while other schemes are added when the concentration or emission of a corresponding species in a certain scheme are greater than zero. Compared with the algorithm computing all chemical interactions, this algorithm is helpful in saving the computation resources on a simple core, while such irregular and unbalanced calculations lack well-structured loops and impede the vectorization of codes. Besides the chemistry sub-schemes mentioned above, CBM-Z uses the other functional branches, e.g. nocturnal and diurnal chemistry,

and they impede the vectorization of the computation. The CBM-Z also contains multiple unconstructed scalar operations. We partially integrated the scalar operations by using indirect indexes to construct loops for vectorization (Wang et al., 2017). However, this method required significant efforts, and it only reconstructed a limited number of scalar operations. The CBM-Z module still contains many scalar operations. With multi-level control flow divergences and many scalar calculations, it is not feasible to perform automatic vectorization with an Intel compiler.

Fortunately, contiguous model grids may have similar chemical processes in air quality simulation, which provides the opportunity to integrate the grids with similar or the same chemical processes to implement vectorization to calculate the processes of multiple grids simultaneously. The following section introduces the details about integrating the chemistry sub-schemes to implement the vectorization.

2.2 Algorithm Description

The new generation Intel CPU (e.g., Skylake) and Intel MIC chips are equipped with the AVX-512 or more advanced vectorization instructions, which supports a maximum of eight double precision and sixteen single precision operations with 512-bit-wide vector registers. It is critical to peak performance of the next generation CPUs and MICs to fully reach the potential of the AVX-512(Mielikainen et al., 2014). As mentioned in section 2.1, the automatic vectorization using a compiler is impeded by the features of CBM-Z, and the common manual measures including constructing loops, avoiding the loop/data dependence and aligning the data with directives need to further vectorize CBM-Z. On the other hand, to implement the vectorization of the module, the general design allowed the CBM-Z module to handle multiple grids in one citing cycle, and the functions in CBM-Z were reconstructed by adding a regular loop for these grids. Subsequently, these loops can be vectorized to implement the fine-grain parallel on a VPU.

All of the model grids are distributed to multiple cores using a Message Passing Interface (MPI) and OpenMP, which is a type of coarse-grain parallelization. Our goal is to implement fine-grain parallelization based on the SIMD, and the grids that are distributed to a specific processor operate in parallel using the VPUs on each core. As shown in Figure 2, the calling method of the CBM-Z module changes from calculating one model grid calculation at a time to multiple model grids at the same time. The step length (VLEN in Figure 2) of the loops represents the number of the grids operated simultaneously, and it is determined by the length of the vector register. The VLEN was set to 16 since the 512-bit-wide vector of AVX-512 can support 16 single-precision operations at the same time. Using this framework, the functions in CBM-Z construct an extra loop to manage the point number dimension, and the corresponding variables require an extra dimension to store the information of multiple grids. Using the structure with an extra loop, it was easier to implement the vectorization. Meanwhile, to avoid multiple remaining points which cannot satisfy the VLEN, we set a common variable array, ***pmask***(VLEN) as shown in Figure 2, to store the availability label of the model grids. When the number of remaining grids did not reach VLEN, the corresponding ***pmask*** value of excessive grids was set to False to mask these grids in the calculation. Furthermore, the latitude and longitude dimensions loop were merged, from nested loops to a single loop, to reduce the number of unavailable points as shown in Figure 2. Achieving such a large-scale vectorization also requires the balance of the calculation processes, but the calculation

branches in CBM-Z are an obstacle to balancing the calculation. Therefore, the branches in CBM-Z should be taken into consideration in constructing the loops, especially the chemical schemes chosen in Table 1. As mentioned in section 2.1, the contiguous model grids may have similar chemical processes in the atmosphere. This provides an opportunity to integrate the sub-schemes by masking the heterogeneous model grids, and this type of masking operation can be used in the functions ***GasRateConstants*** and ***ODEsolver*** (Figure 1). Figure 3 shows the flowchart for masking the model grids to satisfy the vectorization of the grid array. The VLEN number grids contain an array to perform the operation simultaneously, and the variable ***pmask*** signed the valid grids. Meanwhile, the variable ***iregime*** described in Table 1 and representing the combination of sub-schemes, is used to determine whether the model grid must perform the subsequent operation or not. The grids with the same property or calculation are kept by setting the variable ***bmask*** to True. The COM and HET schemes are common for all grids, and the mask operation for COM and HET schemes only determine the availability of the grids. As shown in Figure 3, for the URB, BIO and MAR schemes, the ***iregime*** value and ***pmask*** are both used to filter the heterogeneous grids and the ***bmask*** stores the results. To improve the efficiency of vectorization, the ***bmask*** does not prevent the calculation of heterogeneous grids but prevents the calculation results from copying back to the return value. Thus, all computations are performed on all grids, but only the results of the valid grids are returned. This improves the utilization of data as well as the efficiency of vectorization. Because of the independence of the grids, the computation process of VLEN arrays are independent and satisfy the requirement of vectorization, and the corresponding directives were added to declare the independence of the arrays and force the compiler to perform the data alignment and vectorization after the reconstruction of the codes. Overall, by constructing the loops, the computations of the independent grids were integrated with the fine-level parallel implementation through the SIMD. In addition, the efficiency of such algorithms is linearly improved with the development of the width of the vector in the VPU.

3 Test Results

The validation and evaluation of the improvement of the new method were conducted using the box model of CBM-Z. Two cases were used for testing. One was a single point case with all species to validate the outputs of the model, and the other was a 10-hour simulation with 160*148*20 grids to test the performance. The initial values of the single point case were showed in Table S1. The meteorological conditions were constant and emissions were set to zero to test the error of the algorithms. The time step-size was 5 seconds for two cases. The case used for validating output the simulation results every 5 minutes, while the performance test did not open the output function to isolate the disturbance of I/O. We tested the baseline and optimized model on two different platforms of CPU and KNL, and the time of the computation portion was counted using the ***system_clock*** function.

Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

3.1 Benchmark Platform Description

The CPU and KNL platforms were used for testing the codes. The CPU platform was a 2-socket CPU node containing two 2.3 GHz 18-core Intel Xeon E5-2697 V4 processors, and its operating system was CentOS release 6.7. The KNL node contained one 1.40 GHz 68-core Intel Xeon Phi 7250 processor, and its operating system was Red Hat Enterprise Linux 7.2. The Intel Xeon Phi 7250 had the better vectorization ability with AVX-512 vector instructions compared with the Intel Xeon E5-2697 V4 with AVX2 vector instructions. The codes were all compiled using the Intel FORTRAN Compiler 2017 update 4, and the compile flags are shown in Table 2. The corresponding flags for vectorization (e.g., –xCORE-AVX2, -align array64byte) were adopted.

3.2 Results Validation

The test case for validation was a one-point simulation including all reactions and species. The simulation results of all species were output every five minutes with the meteorological conditions updated. We validated the chemical species including ozone (O_3), nitrogen dioxide (NO_2), nitrogen monoxide (NO), hydrogen peroxide (H_2O_2), sulfur dioxide (SO_2), and sulfuric acid (H_2SO_4). These species are relatively representative and suitable for validating whether the optimization significantly changed the simulated results or not. Figure 4 and Figure 5 show the time-series and the scatter of the simulated concentrations of the species above from the baseline (BASE) and optimized (OPT) version model. The time-series lines are completely coincident and most of the points in Figure 5 are on the diagonal. The average relative errors from the digital reservation were in an acceptable range of lower than 0.05%. In addition, with the -O0 parameter set, the difference between the two model outputs decreased, which suggests no logical and artificial errors of the optimized codes.

3.3 Performance Test

The case with 160*148*20 grids was used as the benchmark case to test the performance. The platform configuration was introduced in section 3.1. Both the baseline and the optimized version of CBM-Z contained the same 76 species. The computational time of the baseline version on a single core of E5-2697 V4 CPU was considered as the benchmark time to evaluate the speedup of the optimized version. As shown in Table 3, the baseline performance was 539.87 seconds on the CPU platform with one core and 4481.10 seconds with one core on the KNL platform, while the baseline version of CBM-Z showed the worst performance with the single KNL core because of the low frequency. The optimized version CBM-Z consumed 127.07 seconds with one core on the CPU and 258.56 seconds with one core on the KNL. Compared with the benchmark time on the CPU, the corresponding speedup reached 4.24x and 2.08x respectively. The speedup between the baseline and the optimized CBM-Z on the single KNL core was approximately 17.33x. The speedups on the KNL and CPU both indicate a significant improvement in vectorization, which was reflected in the single core test. The speedup on the KNL was more significant than on the CPU in this test because of the wider vector, and other micro-architecture optimizations also contributed to the speedup on the KNL. Meanwhile, based on the HPC Performance Test from the Intel VTune tools on the CPU, the single

precision giga-floating point operations calculated per second (GFLOPS) increased from 5.62 to 13.92, and the vector capacity usage improved from 32.1% in the baseline CBM-Z to 100% in the optimized CBM-Z. This result suggests that all floating-point instructions in CBM-Z were vectorized with the full vector capacity.

We also tested the parallel version of CBM-Z with the MPI and OpenMP separately, and the results showed that the MPI and OpenMP version of CBM-Z had 113.42x speedup and 118.13x speedup on the CPU, respectively. For the KNL, the speedup reached 170.31x by using MPI and 179.95x by using OpenMP, which was approximately 50～52% faster than those on the 2-socket CPU platform. The combination of the fine-grain vectorization and the coarse-grain parallelization of OpenMP/MPI resulted in a significant performance improvement on the new generation processors. The enhancement of the vectorization performance may be the key to fully using the new generation processors equipped with advanced and wider vectors, which can be significant to fully use the new MIC architecture processors such as KNL.

4 Conclusion and Discussion

A new framework was designed for helping the chemical kinetics kernel CBM-Z to adapt to the next generation processes by improving its vectorization. Through packing the multiple spatial points, the optimized CBM-Z module handled these grids simultaneously. The functions in the original CBM-Z were reconstructed with loops, which provided the opportunity to implement the fine-grain level parallelization of vectorization. Meanwhile, we masked the heterogeneous grids to integrate the chemistry sub-schemes in the CBM-Z to perform the calculation of multiple grids simultaneously. Since the contiguous grids had similar chemistry processes, the impact of these process on performance was largely limited, and the codes were highly vectorized.

The platform equipped with the CPU (Intel E5-2697 V4) and KNL (Intel Xeon Phi 7250) were used to test the performance respectively. The validation test ensured the reliability of our optimization on the model results, and the discrepancies of all diagnostic chemical species caused by digital reservation were lower than 0.05%. Based on the HPC Performance Test from the Intel Vtune tools on the CPU, the GFLOPS increased from 5.62 to 13.92, and the vector capacity usage improved from 32.1% in baseline CBM-Z to 100% in the optimized CBM-Z. The test using the single core showed that the vectorization optimization led to a 4.24x speedup on the CPU and a 17.33x speedup on the KNL. It highlights the importance of vectorization to the KNL platform. Meanwhile, we also tested the MPI and OpenMP version CBM-Z. The speedups on the CPU reached 113.42x and 118.13x by using MPI and OpenMP respectively, and the corresponding speedups on KNL reached 170.31x and 179.95x. The optimized CBM-Z showed 50~52% higher performance on the KNL platform than on the 2-socket CPU platform due to the better parallel ability and vectorization ability of the AVX-512 of KNL.

Besides the CBM-Z chemical scheme, this algorithm is also suitable for models with a similar code structure to improve its vectorization. In addition, in this study, CBM-Z was treated as an example to describe this simple optimization strategy to implement the optimization on new generation processors, which emphasizes the importance of vectorization. However, some specific strategies should also be considered before adoption. The optimizing methods such as constructing loops from the

discrete scalar calculations as described in Wang et al. (2017), would diminish the readability of the source code by using a mediate or indirect index and could cause problems to the following developers. Therefore, it is essential to write annotations when the developers are doing the optimization. Maintaining the original code and controlling the compile process allow the next developer to understand the code.

This work is a part of the parallel optimization project for GNAQPMS model, and our future work will focus on the whole model optimization. The scalability of the GNAQPMS model and I/O bottleneck is the focus of our following work.

Code Availability.

The source code of the baseline and optimized version CBM-Z box model, including OpenMP and MPI versions, is available online via ZENODO (https://doi.org/10.5281/zenodo.1161576).

Acknowledgements.

The National Key R&D Program of China (2017YFC0209805 and 2016YFB0200800), the CAS Information Technology Program (XXH13506-302), the National Natural Science Foundation of China (41305121) and the Fundamental Research Funds for the Central Universities funded this work.

References

Chang, J. S., Brost, R. A., Isaksen, I. S. A., Madronich, S., Middleton, P., Stockwell, W. R., and Walcek, C. J.: A three-dimensional Eulerian acid deposition model: Physical concepts and formulation, Journal of Geophysical Research Atmospheres, 92, 14681-14700, 1987.

Chen, H. S., Wang, Z. F., Li, J., Tang, X., Ge, B. Z., Wu, X. L., Wild, O., and Carmichael, G. R.: GNAQPMS-Hg v1.0, a global nested atmospheric mercury transport model: model description, evaluation and application to trans-boundary transport of Chinese anthropogenic emissions, Geosci. Model Dev., 8, 2857-2876, 10.5194/gmd-8-2857-2015, 2015.

Feng, F., Wang, Z., Li, J., and Carmichael, G. R.: A nonnegativity preserved efficient algorithm for atmospheric chemical kinetic equations, Applied Mathematics & Computation, 271, 519-531, 2015.

Gao, M., Carmichael, G. R., Wang, Y., Saide, P. E., Yu, M., Xin, J., Liu, Z., and Wang, Z.: Modeling study of the 2010 regional haze event in the North China Plain, Atmos. Chem. Phys., 16, 1673-1691, 10.5194/acp-16-1673-2016, 2016.

Grell, G. A., Peckham, S. E., Schmitz, R., McKeen, S. A., Frost, G., Skamarock, W. C., and Eder, B.: Fully coupled "online" chemistry within the WRF model, Atmospheric Environment, 39, 6957-6975, https://doi.org/10.1016/j.atmosenv.2005.04.027, 2005.

Gurjar, B. R., Ravindra, K., and Nagpure, A. S.: Air pollution trends over Indian megacities and their local-to-global implications, Atmospheric Environment, 142, 475-495, https://doi.org/10.1016/j.atmosenv.2016.06.030, 2016.

Li, J., Wang, Z., Zhuang, G., Luo, G., Sun, Y., and Wang, Q.: Mixing of Asian mineral dust with anthropogenic pollutants over East Asia: a model case study of a super-duststorm in March 2010, Atmospheric Chemistry & Physics, 12, 7591-7607, 2012.

Linford, J. C.: Multi-core acceleration of chemical kinetics for simulation and prediction, High Performance Computing Networking, Storage and Analysis, Proceedings of the Conference on, 2013, 7,

Mielikainen, J., Huang, B., and Huang, A. H. L.: Intel Xeon Phi accelerated Weather Research and Forecasting (WRF) Goddard microphysics scheme, Geosci. Model Dev. Discuss., 2014, 8941-8973, 10.5194/gmdd-7-8941-2014, 2014.

San José, R., Pérez, J. L., Balzarini, A., Baró, R., Curci, G., Forkel, R., Galmarini, S., Grell, G., Hirtl, M., Honzak, L., Im, U., Jiménez-Guerrero, P., Langer, M., Pirovano, G., Tuccella, P., Werhahn, J., and Žabkar, R.: Sensitivity of feedback effects in CBMZ/MOSAIC chemical mechanism, Atmospheric Environment, 115, 646-656, https://doi.org/10.1016/j.atmosenv.2015.04.030, 2015.

Seinfeld, J. H., and Pandis, S. N.: Atmospheric Chemistry and Physics: From Air Pollution to Climate Change, 2nd Edition, 2012.

Sodani, A., Gramunt, R., Corbal, J., Kim, H. S., Vinod, K., Chinthamani, S., Hutsell, S., Agarwal, R., and Liu, Y. C.: Knights Landing: Second-Generation Intel Xeon Phi Product, IEEE Micro, 36, 34-46, 2016.

Wang, H., Chen, H., Wu, Q., Lin, J., Chen, X., Xie, X., Wang, R., Tang, X., and Wang, Z.: GNAQPMS v1.1: accelerating the Global Nested Air Quality Prediction Modeling System (GNAQPMS) on Intel Xeon Phi processors, Geosci. Model Dev., 10, 2891-2904, 10.5194/gmd-10-2891-2017, 2017.

Wang, Z., Maeda, T., Hayashi, M., Hsiao, L. F., and Liu, K. Y.: A Nested Air Quality Prediction Modeling System for Urban and Regional Scales: Application for High-Ozone Episode in Taiwan, Water, Air, and Soil Pollution, 130, 391-396, 10.1023/A:1013833217916, 2001.

Wu, Q. Z., Xu, W. S., Shi, A. J., Li, Y. T., Zhao, X. J., Wang, Z. F., Li, J. X., and Wang, L. N.: Air quality forecast of PM10 in Beijing with Community Multi-scale Air Quality Modeling (CMAQ) system: emission and improvement, Geosci. Model Dev., 7, 2243-2259, 10.5194/gmd-7-2243-2014, 2014.

Xu, S., Huang, X., Oey, L. Y., Xu, F., Fu, H., Zhang, Y., and Yang, G.: POM.gpu-v1.0: a GPU-based Princeton Ocean Model, Geosci. Model Dev., 8, 2815-2827, 10.5194/gmd-8-2815-2015, 2015.

Zaveri, R. A., and Peters, L. K.: A new lumped structure photochemical mechanism for long-scale applications, Journal of Geophysical Research Atmospheres, 104, 30387-30415, 1999.

Zhang, Q., Jiang, X., Tong, D., Davis, S. J., Zhao, H., Geng, G., Feng, T., Zheng, B., Lu, Z., Streets, D. G., Ni, R., Brauer, M., van Donkelaar, A., Martin, R. V., Huo, H., Liu, Z., Pan, D., Kan, H., Yan, Y., Lin, J., He, K., and Guan, D.: Transboundary health impacts of transported global air pollution and international trade, Nature, 543, 705-709, 10.1038/nature21712 http://www.nature.com/nature/journal/v543/n7647/abs/nature21712.html - supplementary-information, 2017.

Zimmermann, J., and Poppe, D.: A Supplement for the RADM2 Chemical Mechanism: The Photooxidation of Isoprene, Atmospheric Environment, 30, 1255–1269, 1994.

Table 1. The possible values of iregime and the combination of chemical schemes.

iregime	1	2	3	4	5	6
Sub-schemes	COM	COM	COM	COM	COM	COM
	HET	HET	HET	HET	HET	HET
		URB	URB		URB	URB
			BIO			BIO
				MAR	MAR	MAR

Table 2. Compile flags of the different version of CBM-Z.

Version of CBM-Z	Intel Compiler Flags
Baseline	-O3 –g –xCORE-AVX2 –fp-model fast=1
Optimized codes on CPU	-O3 –fp-model fast=1 –xCORE-AVX2 –align array64byte –qopenmp
Optimized codes on KNL	-O3 –fp-model fast=1 –xMIC-AVX512 –align array64byte –qopenmp

Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

Table 3. The performance tests of the baseline and optimized codes on the CPU and KNL platforms.

	Processor Type	Number of Cores (Frequency of Cores)	Time (s)	Speedup
Baseline CBM-Z	CPU	1 (2.3 GHz)	539.87	1
	KNL	1 (1.4 GHz)	4481.10	0.12
Optimized CBM-Z	Vectorization			
	CPU	1 (2.3 GHz)	127.07	4.24
	KNL	1 (1.4 GHz)	258.56	2.08
	MPI with Vectorization			
	2-socket CPU	32 (2.3 GHz)	4.76	113.42
	KNL	68 (1.4 GHz)	3.17	170.31
	OpenMP with Vectorization			
	2-socket CPU	32 (2.3 GHz)	4.57	118.13
	KNL	68 (1.4 GHz)	3.00	179.95

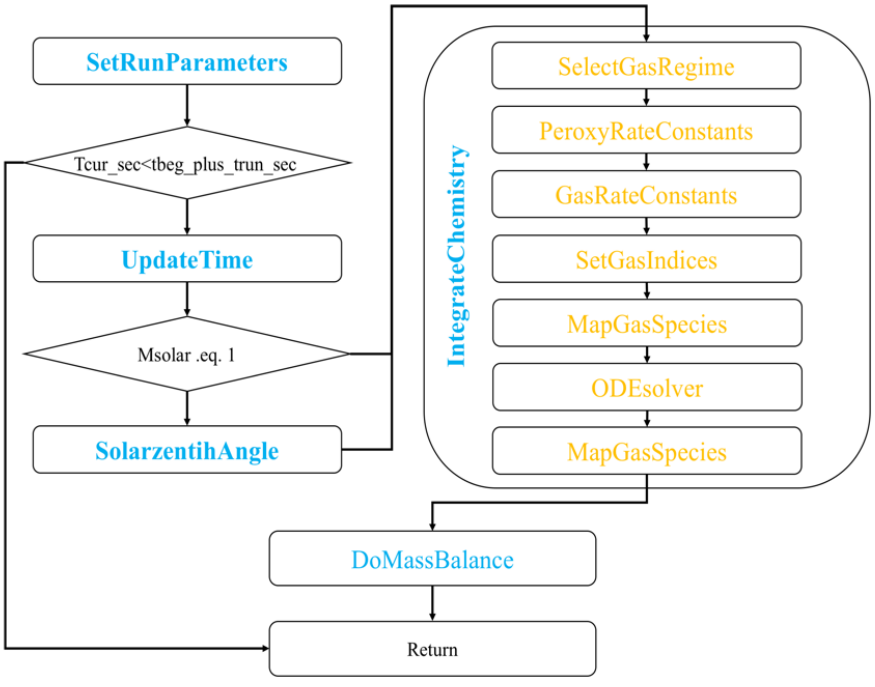

Figure 1. The framework of the CBM-Z gas-phase chemistry module. The functions in the yellow font represent the inner function of IntegrateChemistry.

Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

Geoscientific Model Development Discussions

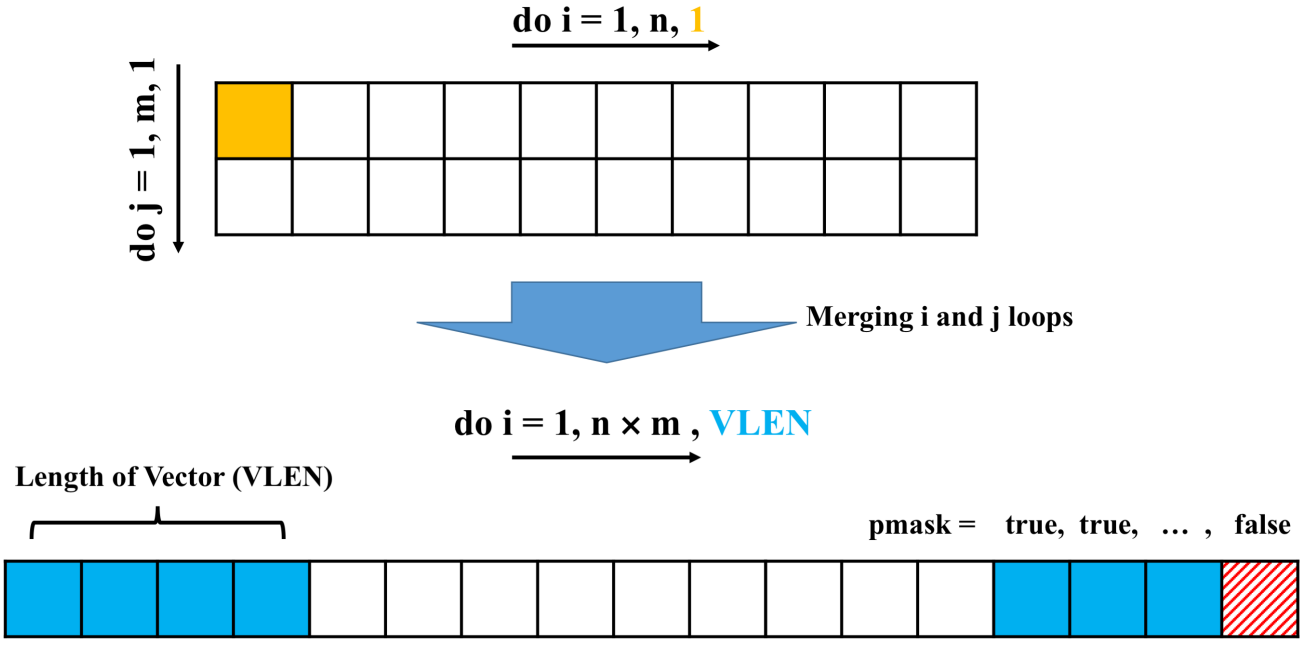

Figure 2. A schematic diagram of the changes of the calling method of CBM-Z. The calling method of the CBM-Z module changes from calculating one model grid calculation at a time to multiple model grids at the same time. The VLEN represents the number of points operated simultaneously, which is determined by the length of the register in the Vector Processing Unit (VPU). The i and j loops, equaled latitude and longitude loops, were merged to construct one vector to reduce the number of unfilled vector.

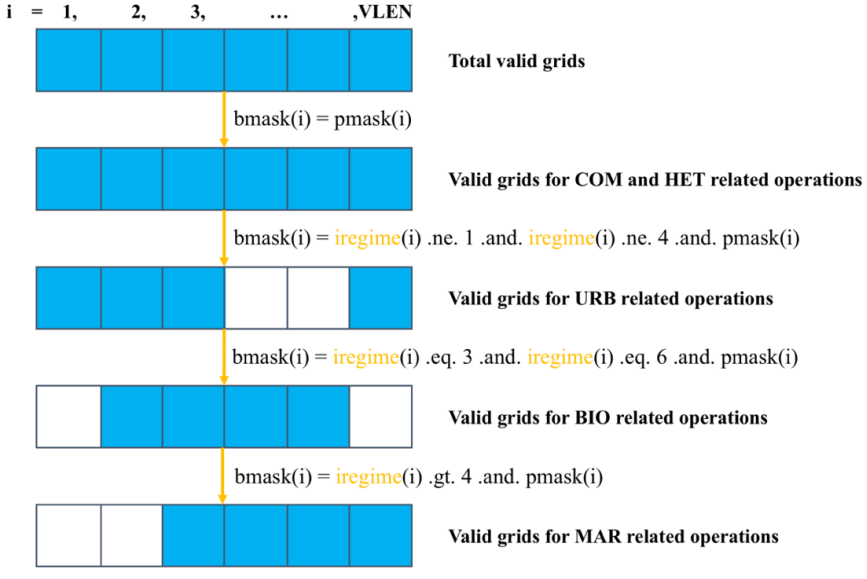

Figure 3. The flowchart shows the way to mask the heterogeneous girds to integrate grids to perform the vectorization operations according to the iregime values.

Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

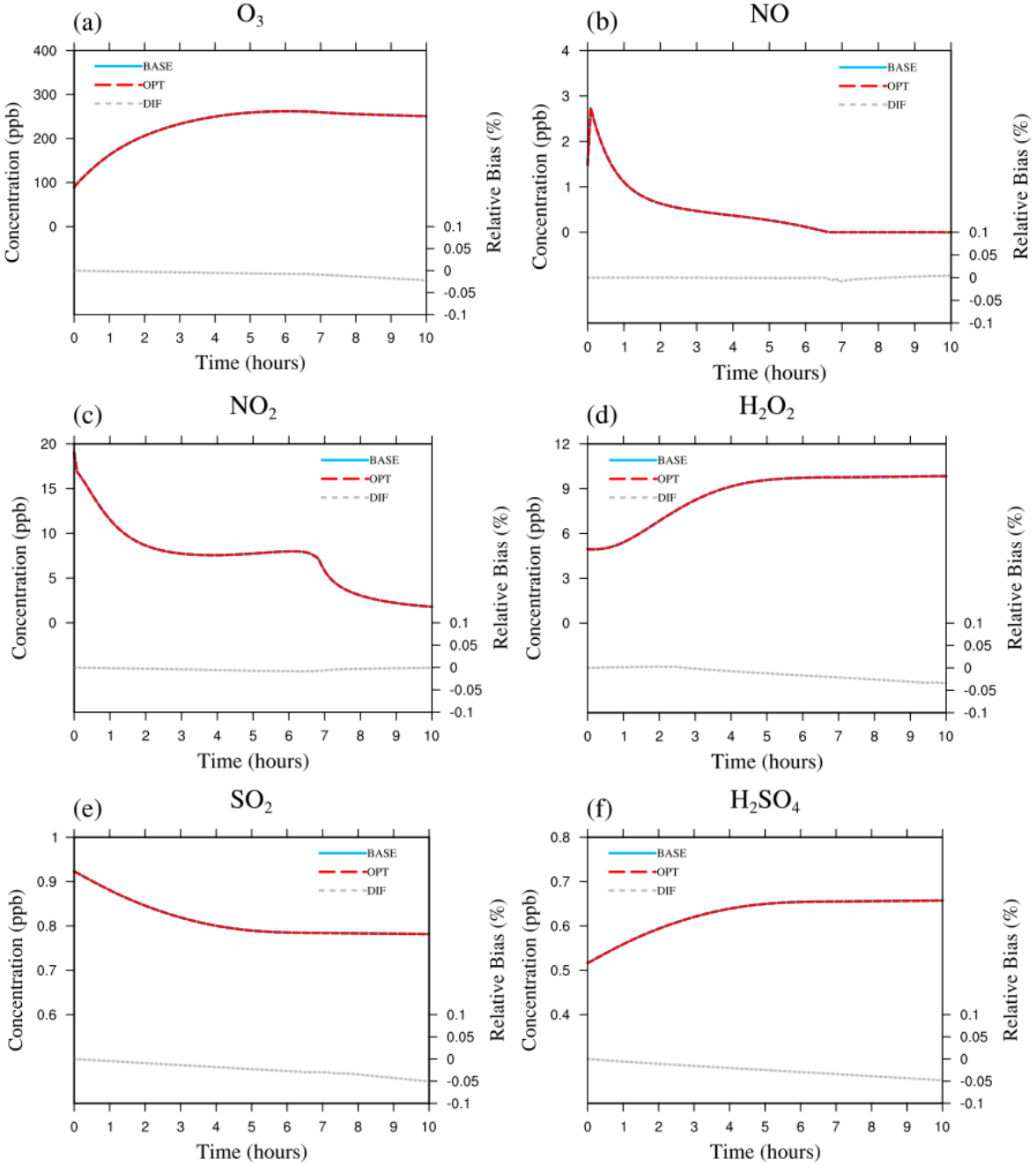

Figure 4. Comparison of the time-series concentrations of O_3, NO, NO_2, H_2O_2, SO_2 and H_2SO_4 ((a)-(f)) from the baseline and optimized CBM-Z simulation.

Geosci. Model Dev. Discuss., https://doi.org/10.5194/gmd-2018-42
Manuscript under review for journal Geosci. Model Dev.
Discussion started: 24 April 2018

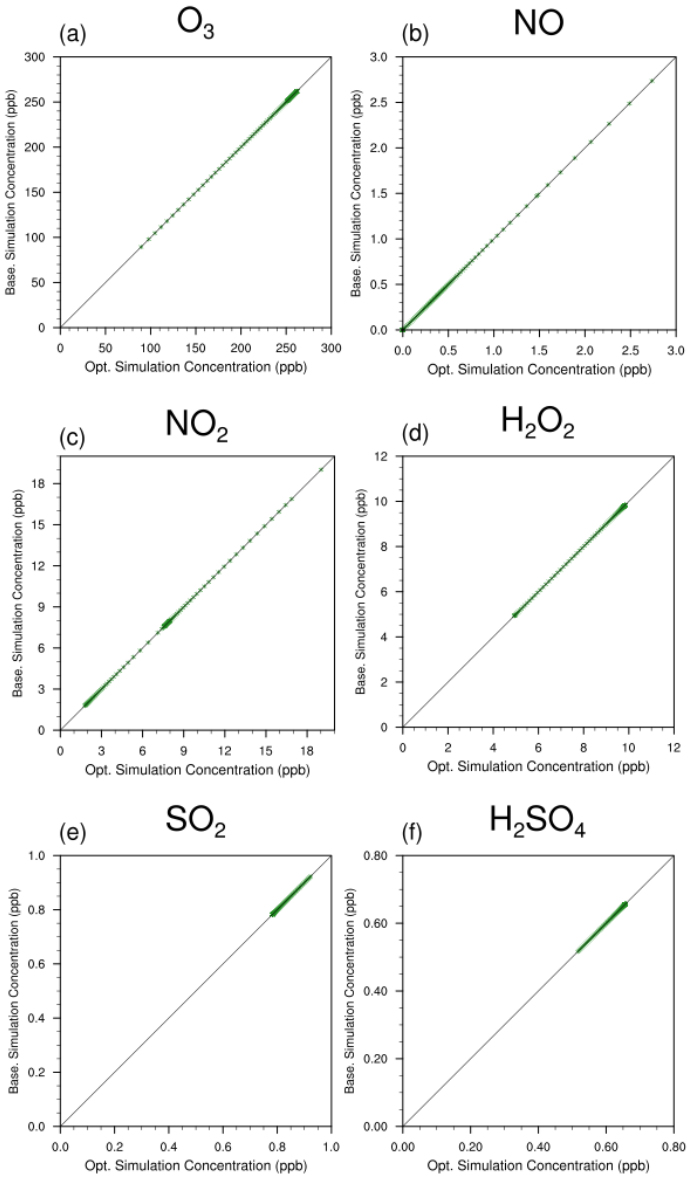

Figure 5. The scatter diagram of the results from the baseline and optimized version codes, respectively.