Scalability Improvements in the NASA Goddard Multiscale Modeling Framework for Tropical Cyclone Climate Studies

Bo-Wen Shen[1,2], Wei-Kuo Tao[1], Jiun-Dar Chern[1,3], Robert Atlas[4], K. Palaniappan[5]
[1]Laboratory for Atmospheres
NASA Goddard Space Flight Center
Greenbelt, MD 20771, USA
bo-wen.shen-1@nasa.gov

[2]Earth System Science Interdisciplinary Center
University of Maryland, College Park
College Park, MD 20742, USA

[3]Goddard Earth Science and Technology Center
University of Maryland, Baltimore Country
Baltimore, MD 21228

[4]NOAA Atlantic Oceanographic and Meteorological Laboratory
4301 Rickenbacker Causeway
Miami, FL 33149

[5]Department of Computer Science
University of Missouri-Columbia
Columbia, MO 65211-2060

Abstract

A current, challenging topic in tropical cyclone (TC)[1] research is how to improve our understanding of TC inter-annual variability and the impact of climate change on TCs. Paired with the substantial computing power of the NASA Columbia supercomputer, the newly-developed multi-scale modeling framework (MMF) [1] shows potential for the related studies. The MMF consists of two NASA state-of-the-art modeling components, including the finite-volume General Circulation Model (fvGCM) and the Goddard Cumulus Ensemble model (GCE). For TC climate studies, the MMF's computational issues (e.g., limited scalability) need to be addressed. By introducing a meta grid system, we integrate the GCEs into a meta-global GCE, and apply a 2D domain decomposition in this grid-point space. A prototype parallelism implementation shows very promising scalability, giving a nearly linear speedup as the number of CPUs is increased from 30 to 364. This scalability improvement makes it more feasible to study TC climate. Future work on further model improvement will be also discussed.

[1] Depending on their location, TCs are referred to by other names, such as hurricane (in the Atlantic region), typhoon (in the West Pacific region), tropical storm, cyclonic storm, and tropical depression.

1. Introduction

Studies in TC inter-annual variability and the impact of climate change (e.g., global warming) on TCs have received increasing attention [2], particularly due to the fact that 2004 and 2005 were the most active hurricane seasons in the Atlantic while 2006 was not as active as predicted. Thanks to recent advancements in numerical models and supercomputer technology, these topics can be addressed better than ever before.

Earth (atmospheric) modeling activities have been conventionally divided into three major categories based on scale separations: synoptic-scale, meso-scale, and cloud (micro)-scale. Historically, partly due to limited access to computing resources, TC climate has been studied mainly with general circulation models (GCMs) [3] and occasionally with regional mesoscale models (MMs). The former have the advantage of simulating global large-scale flow, while the latter make it possible to simulate realistic TC intensity and structure with fine grid spacing. However for TC climate studies, the resolutions used in GCMs and MMs were still too coarse to resolve small-scale convective motion, and therefore "cumulus parameterizations" (CPs) were required to emulate the effects of unresolved subgrid-scale motion. Because the development of CPs has been slow, their performance is a major limiting factor in TC simulations.

Cloud-resolving models (CRMs) have been extensively developed to accurately represent non-hydrostatic cloud-scale convection and its interaction with environmental flows, aimed at improving TC prediction and advancing the development of CPs. Recently, an innovative approach that applies a massive number of CRMs in a global environment has been proposed and used to overcome the CP deadlock in GCMs [1,4]. This approach is called the multiscale modeling framework (MMF) or super-parameterization, wherein a CRM is used to replace the conventional CP at each grid point of a GCM. Therefore, the MMF has the combined advantages of the global coverage of a GCM and the sophisticated microphysical processes of a CRM and can be viewed as an alternative to a global CRM. Currently, two MMFs with different GCMs and CRMs have been successfully developed at Colorado State University and NASA Goddard Space Flight Center (GSFC), and both have produced encouraging results in terms of a positive impact on simulations of large-scale flows via the feedback of explicitly resolved convection by CRMs. Among them is the improved simulation of the Madden-Julian Oscillation (MJO) [1], which could potentially improve long-term forecasts of TCs through deep convective feedback. However, this approach poses a great computational challenge for performing multi-decadal runs to study TC climate, because nearly 10,000 copies of the CRM need to run concurrently. These tremendous computing requirements and the limited scalability in the current Goddard MMF restrict the GCM's resolution to about 2 degree (~220km), which is too coarse to capture realistic TC structure. In this report, computational issues and a revised model coupling approach will be addressed with the aim of improving the Goddard MMF's capabilities for TC climate studies.

2. The NASA Columbia Supercomputer and Goddard MMF

In late 2004, the Columbia Supercomputer [5] came into operation with a theoretical peak performance of 60 TFLOPs (trillion floating-point operations per second) at the NASA Ames Research Center (ARC). It consists of twenty 512-cpu nodes, which give 10,240 CPUs and 20 tera-bytes (TB) of memory. Columbia achieved a performance of 51.9 TFLOPs with the LINPACK (Linear Algebra PACKage) benchmark and was ranked second on the TOP500 list in late 2004. The cc-NUMA (cache-coherence non-uniform memory access) architecture supports up to 1 TB shared memory per node. Nodes are connected via a high-speed InfiniBand interconnect, and each node can be operating independently. These unique features enable complex problems to be resolved with large-scale modeling systems.

The Goddard MMF is based on the NASA Goddard finite-volume GCM (fvGCM) and the Goddard Cumulus Ensemble model (GCE). While the high-resolution fvGCM has shown remarkable capabilities in simulating large-scale flows and thus hurricane tracks [6-9], the GCE is well known for its superior performance in representing small cloud-scale motions and has been used to produce more than 90 referreed journal papers [10,11]. In the MMF, the fvGCM is running at a coarse (2^{o}x2.5^{o}) resolution, and 13,104 GCEs are "embedded" in the fvGCM to allow explicit simulation of cloud processes in a global environment. Currently, only thermodynamic feedback between the fvGCM and the GCEs is implemented. The time step for the individual 2D GCE is ten seconds, and the fvGCM-GCE coupling interval is one hour at this resolution. Under this configuration, 95% or more of the total wall-time for running the MMF is spent on the GCEs. Thus, wall-time could be significantly reduced by efficiently distributing the large number of GCEs over a massive number of processors on a supercomputer.

Over the past few years, an SPMD (single program multiple data) parallelism has been implemented in both the fvGCM and GCE with good parallel efficiency separately [12,13]. Therefore, in addition to the massive number of GCEs that need to be coupled, different parallelisms in these two models make coupling very challenging. In the following sections, both the GCE and fvGCM are introduced as well as a revised strategy for coupling these model components.

2.1 The Goddard Cumulus Ensemble model

Over the last two decades, the Goddard Cumulus Ensemble model (GCE) has been developed in the mesoscale dynamics and modeling group, led by Dr. W.-K. Tao, at NASA Goddard Space Flight Center. The GCE has been well tested and continuously improved. The model's main features were described in detail in [14,15], and its recent improvements were documented in [10,11]. Typical model runtime configurations are (a) (256, 256) grid points in the (x, y) directions with a grid spacing of 1-2 km; (b) 40-60 vertical stretched levels with a model top at 10-50 hPa; (c) open or cyclic lateral boundary conditions; and (d) a time step of 6 or 12 seconds. Figure 1 shows cloud visualization from a high-resolution simulation.

The GCE has been implemented with a 2D domain decomposition using MPI-1 (Message Passing Interface version 1) to take advantage of recent advances in supercomputing power [13]. To minimize the changes in the GCE, implementation was done with a separate layer added for data communication, which preserves all of

the original array indices. Therefore, not only code readability for existing modelers/users but also code portability for computational researchers is maintained. In addition to "efficiency" enhancement, tremendous efforts were made to ensure reproducibility in simulations with different CPU layouts. Without this, it would be difficult for model developers to test the model with new changes and to compare long-term simulations generated with different numbers of CPUs.

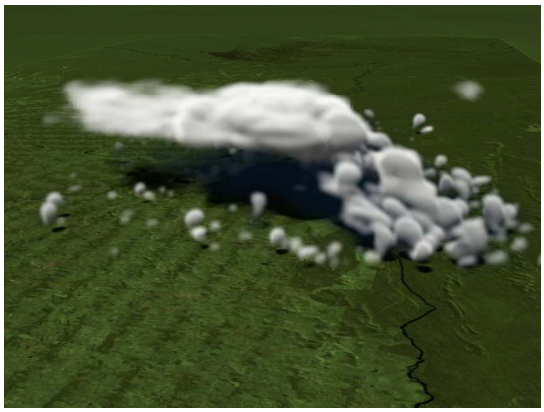

Figure 1: High-resolution cloud simulation of the 23 Feb 1999 TRMM LBA (Large scale Biosphere-Atmosphere Experiment in Amazonia) case with the GCE, which has been implemented with a 2D domain decomposition using MPI-1. A benchmark study shows 99% parallel efficiency with up to 256 CPUs on three different supercomputing platforms, including an HP/Compaq, an IBM-SP Power4, and a SGI Origin 2000 [13].

The scalability and parallel efficiency of the GCE's parallelism implementation was extensively tested on three different supercomputing platforms: an HP/Compaq (HALEM), an IBM-SP Power4, and an SGI Origin 2000 (CHAPMAN). For both anelastic and compressible versions of the GCE, 99% parallel efficiency can be reached with up to 256 CPUs on all of the above machines [13]. Recently, the 3D version of the GCE was ported onto the NASA Columbia supercomputer, and an attempt to scale the model beyond one 512-cpu node is being made, which can be used to help understand the applicability of running massive numbers of 3D GCEs in the MMF environment.

2.2 The finite-volume General Circulation Model

Resulting from a development effort of more than ten years, the finite-volume General Circulation Model (fvGCM) is a unified numerical weather prediction (NWP) and climate model that can run on daily, monthly, decadal, or century time-scales. It has the following major components: (1) finite-volume dynamics [16], (2) physics packages from the NCAR Community Climate Model Version 3 (CCM3) [17], and (3) the NCAR Community Land Model Version 2 (CLM2) [18]. The model was originally designed for climate studies at a coarse resolution of about 2x2.5 degree in the 1990s, and its resolution was increased to 1 degree in 2000 and 1/2 degree in 2002 for NWP [19].

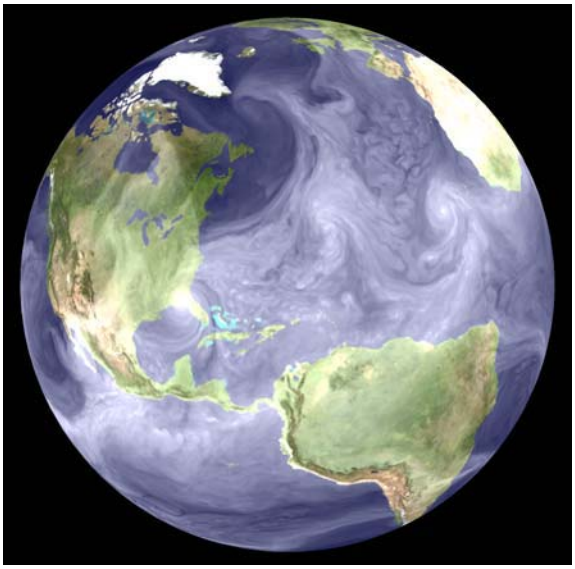

Figure 2: This global view shows total precipitable water from 5-day forecasts initialized at 0000 UTC September 1 2004 with the 1/8 degree fvGCM, which is one of the ultra-high resolution global models. In the Goddard MMF, each of grid points in the fvGCM at a resolution of 2x2.5 degree is running a 2D GCE.

Since 2004, the ultra-high resolution (e.g., 1/8 and 1/12 degree) fvGCM has been deployed on the Columbia supercomputer (Figure 2), showing remarkable TC forecasts.

The parallelization of the fvGCM was carefully designed to achieve efficiency, scalability, flexibility, and portability. Its implementation had a distributed- and shared-memory two-level parallelism, including a coarse grained parallelism with MPI[2] (MPI-1, MPI-2, MLP, or SHMEM) and fine grained parallelism with OpenMP [13]. The model's dynamics, which require a lot of inter-processor communication, have 1D MPI/MLP/SHMEM domain decomposition in the y direction and OpenMP multithreading in the z direction. One of the prominent features in the implementation is to allow multi-threaded

[2] To simplify discussion in this article, the term "MPI" used along with the fvGCM will be referred to as any one of MPI-1/MPI-2/MLP/SHMEM communication paradigms.

data communication. The physical part was parallelized with the 1D domain decomposition in the y direction inherited from the dynamics part and further enhanced with an OpenMP loop-level parallelism in the decomposed latitudes. CLM2 was also implemented with both MPI and OpenMP parallelism, allowing its grid cells to be distributed among processors. Between dynamical grid cells and land patches, a data mapping (or redistribution) is required.

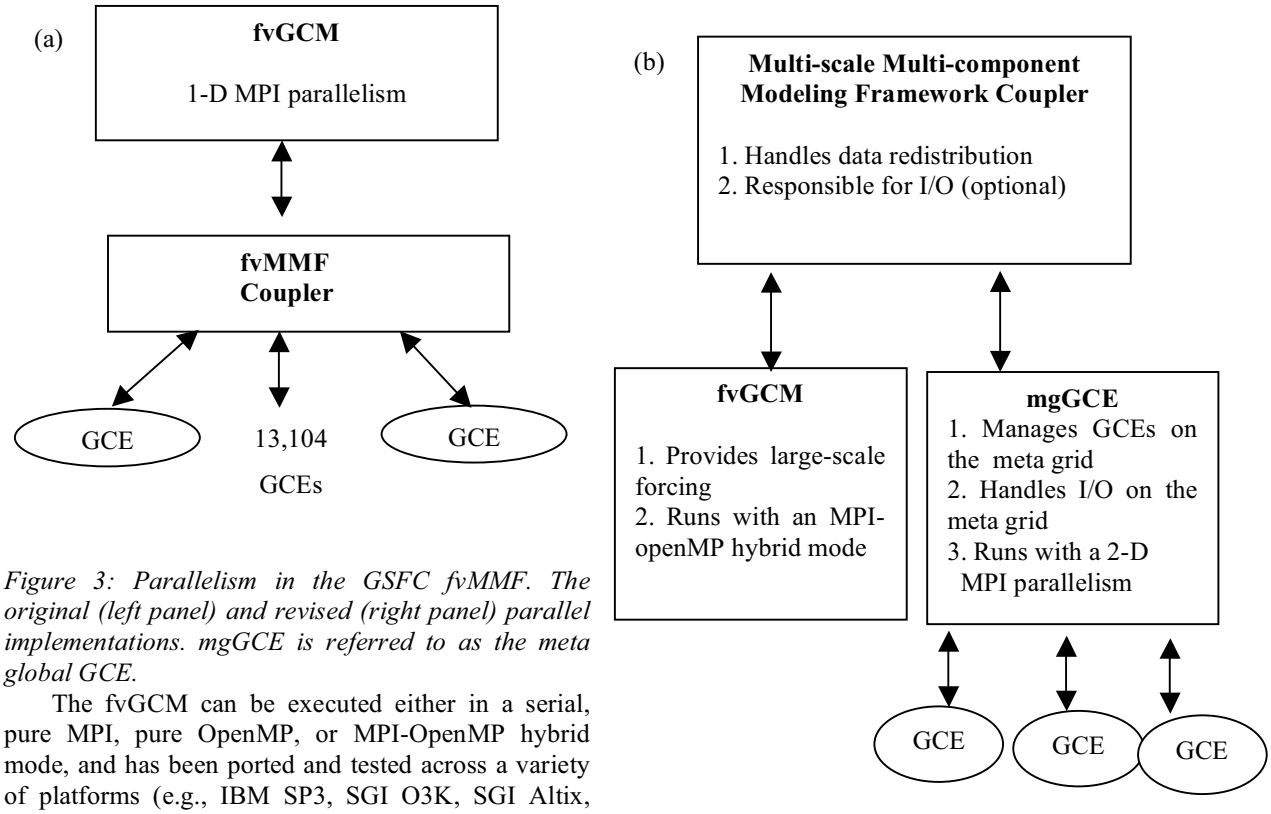

Figure 3: Parallelism in the GSFC fvMMF. The original (left panel) and revised (right panel) parallel implementations. mgGCE is referred to as the meta global GCE.

The fvGCM can be executed either in a serial, pure MPI, pure OpenMP, or MPI-OpenMP hybrid mode, and has been ported and tested across a variety of platforms (e.g., IBM SP3, SGI O3K, SGI Altix, Linux boxes, etc) with different Fortran compilers (e.g., Intel, SGI, IBM, DEC ALPHA, PGI, Lahey, etc). Bit-by-bit reproducibility is ensured on the same platform with different CPU layouts and/or different communication protocols. All of these capabilities speed up model development and tests, thereby making the model very robust. A benchmark with 7-day NWP runs at a 0.5^o resolution[3] on three different platforms: Columbia (SGI Altix 4700), Halem (DEC ALPHA), and Daley (SGI O3K) shows that remarkable scalability was obtained with up to about 250 CPUs [12]. In terms of throughput, the fvGCM could simulate 1110 model days (3+ years) per wall-clock day (days/day) with 240 CPUs on Columbia, 521 days/day with 288 CPUs on Halem, and 308 days/day with 300 CPUs on Daley. Even though these results are not listed for direct comparison due to different interconnect and CPU technologies (e.g., different CPU's clock speeds and cache sizes, etc), it should be noted that a 20% performance increase on Columbia is obtained with the recent upgrades (e.g., an upgrade to the Altix 4700 from the Altix 3000).

2.3 The Goddard MMF

The Goddard MMF implementation consists of the fvGCM at 2^ox2.5^o resolution and 13,104 GCEs, each of which is embedded in one grid cell of the fvGCM (Figure 3). Since it would require a tremendous effort to implement an OpenMP parallelism into the GCE or to extend the 1D domain decomposition to 2D in the fvGCM, the MMF only inherited the fvGCM's 1D MPI parallelism, though the fvGCM was parallelized with both MPI and OpenMP paradigms. This single-component approach limited the MMF's scalability to 30 CPUs, and thereby posed a challenge for increasing the resolution of the fvGCM and/or extending the GCE's dimension from 2D to 3D. To overcome this difficulty, a different strategic approach is needed to couple the fvGCM and GCEs.

[3] A resolution of $2x2.5^o$ is being used in the fvGCM within the MMF, and 1^o is the target resolution in this study. Thus, 0.5^o should be sufficient for now. Benchmarks at higher resolution (e.g., 0.25^o) are being performed on Columbia and will be documented in a separate study.

4. Discussion on the Enhanced MMF

From a computational perspective, the concept of "embedded GCEs" should be completely forgotten, as it restricts the view on the data parallelism of the fvGCM. Instead, the 13,104 GCEs should be viewed as a *meta global GCE* (mgGCE) in a *meta gridpoint system*, which includes 13,104 grid points. This grid system, which is not tied to any specific grid system, is assumed to be the same as the latitude-longitude grid structure in the fvGCM for convenience. With this concept in mind, each of the two distinct parts (the fvGCM and mgGCE) in the MMF could have its own scaling properties (Figure 3b). Since most of wall-time was spent on the GCEs, we could substantially reduce the wall-time by deploying a highly scalable mgGCE and coupling the mgGCE and the fvGCM using an MPMD (multiple programs multiple data) parallelism.

Data parallelism in the mgGCE indeed becomes a task parallelism, namely distributing 13,104 GCEs among processors. Because cyclic lateral boundary conditions are used in each GCE, the mgGCE has no ghost region in the meta grid system and can be scaled "embarrassingly" with a 2D domain decomposition. For the coupled MMF, which has major overhead only in data redistribution (or data regridding) between the fvGCM and the mgGCE, its scalability and performance will depend mainly on the scalability and performance of the mgGCE and the coupler, which is the interface between the fvGCM and mgGCE. Under this current definition, a grid inside each GCE, running at one meta grid, becomes a *child grid* (or sub-grid) with respect to the parent (meta) grid (Figure 3b). Since an individual GCE can still be executed with its native 2D MPI implementation in the child grid-point space, this second level of parallelism can greatly expand the number of CPUs. Potentially, the coupled MMF along with the mgGCE could be scaled at a multiple of 13,104 CPUs. Having two different components, this coupled system is also termed a multi-scale multi-component modeling framework in this study.

Another advantage of introducing the mgGCE component is to allow the adoption of the idea of land-sea masks used in a land model. For example, if computing resources are limited, a cloud-mask file can be used to specify limited regions where the GCEs should be running. A more sophisticated cloud-mask implementation in the mgGCE will enable one to choose a variety of GCEs (2D vs. 3D, bulk vs. bin microphysics) depending on geographic location. Thus, computational load balances can be managed efficiently.

To achieve all of the aforementioned functionalities, a scalable and flexible coupler and a scalable parallel I/O module need to be developed. The coupler should be designed carefully in order: (1) to minimize the changes in the GCE and permit it as a stand-alone application or a single element/component in the mgGCE; (2) to seamlessly couple the mgGCE and fvGCM to allow for a different CPU layout in each of these components; (3) to allow the mgGCE to be executed in a global, channel, or regional environment with a suitable configuration in the cloud-mask file. A scalable (parallel) I/O module needs to be implemented in the meta grid-point space, since it is impractical to have the individual GCE to do its I/O.

As a stand-alone model, the mgGCE can be also tested offline with large-scale forcing derived from model reanalysis [e.g., from the Global Forecast System (GFS) at the National Centers for Environmental Prediction (NCEP)] or from high-resolution model forecasts (e.g., the fvGCM). To assure the implementation in the mgGCE is correct, simulations with the mgGCE at a single meta point should be identical to those with a regular GCE. One potential application of the mgGCE is to investigate the short-term evolution of hurricane Katrina's (2005) precipitation by performing simulations driven by the NCEP GFS T382 (~35km) reanalysis data at a 6h time interval. Then, we can extend this approach by replacing the GFS reanalysis with 1/8° fvGCM forecasts at a smaller time interval (see more detailed information about these forecasts in [8]).

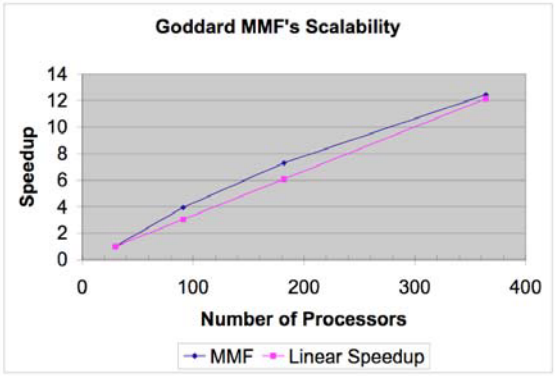

Figure 4: Scalability of the Goddard MMF with a proof-of-concept parallel implementation. This figure shows that a linear speedup is obtained as the number of CPUs increases from 30 to 364. The original MMF could use only 30 CPUs. Further improvement is being conducted.

At this time, a prototype MMF including the mgGCE, fvGCM and coupler has been successfully implemented. The technical approaches are briefly summarized as follows: (1) a master process allocates a shared memory arena for data redistribution between the fvGCM and mgGCE by calling the Unix *mmap* function; (2) the master process spawns multiple (parent) processes with a 1D domain decomposition in the y direction by a series of Unix *fork* system calls; (3) each of these parent processes then forks several child

processes with another 1D domain decomposition along the x direction; (4) data gathering in the mgGCE is done along the x direction and then the y direction; (5) synchronization is implemented with the atomic *__sync_add_and_fetch* function call on the Columbia supercomputer. While steps (1), (2), and (5) were previously used in MLP (multiple level parallelism) [20], this methodology is now extended to the multi-component system.

Figure 4 shows preliminary benchmarks with very promising scalability up to 364 CPUs. Here the speedup is determined by T_{30}/T, where T is the wall time to perform a 5-day forecast with the MMF and T_{30} the time spent using 30 CPUs. The run with 30 CPUs was chosen as a baseline simply because this configuration was previously used for production runs [1]. A speedup of (3.93, 7.28, and 12.43) is obtained by increasing the number of CPUs from 30 to (91, 182, and 364) CPUs, respectively. As the baseline has load imbalance and excessive memory usage in the master process, it is not too surprising to obtain a super-linear speedup. Further analysis of the MMF's throughput indicates that it takes about 164 minutes to finish a 5-day forecast using 364 CPUs, which meets the requirement for performing realtime numerical weather prediction. A yearly simulation would only take 8 days to run with 364 CPUs as opposed to 96 days with 30 CPUs. This makes it far more feasible for studying TC climate. The enhanced coupled model has been used to perform two-year production runs (see details in [21]).

5. Concluding Remarks

Improving our understanding of TC inter-annual variability and the impact of climate change (e.g., doubling CO2 and/or global warming) on TCs brings both scientific and computational challenges to researchers. As TC dynamics involves multiscale interactions among synoptic-scale flows, mesoscale vortices, and small-scale cloud motions, an ideal numerical model suitable for TC studies should demonstrate its capabilities in simulating these interactions. The newly-developed multi-scale modeling framework (MMF) [1] and the substantial computing power by the NASA Columbia supercomputer [5] show promise in pursuing the related studies, as the MMF inherits the advantages of two NASA state-of-the-art modeling components: the fvGCM and 2D GCEs. This article focuses on the computational issues and proposes a revised methodology to improve the MMF's performance and scalability. It has been shown that this prototype implementation can improve the MMF's scalability substantially without the need of major changes in the fvGCM and GCEs.

To achieve these goals, the concept of a meta grid system was introduced, grouping a large number of GCEs into a new component called the mgGCE. This permits a component-based programming paradigm to be used to couple the fvGCM and mgGCE. A prototype MMF is then implemented for data redistribution between these two components. This revised coupled system is also termed a multiscale multicomponent modeling framework as both the fvGCM and mgGCE are separate components with their own parallelism. This proof-of-concept approach lays the groundwork for a more sophisticated modeling framework and coupler to solve unprecedentedly complex problems with advanced computing power. For example, the cloud-mask idea associated with the mgGCE will enable GCEs to run with a variety of choices, including different dimensions (2D vs. 3D) and different microphysical packages (e.g., bulk or bin). The next step is to conduct TC climate studies by performing long-term MMF simulations with a channel mgGCE and 1^{o}x1.25^{o} fvGCM. A global channel ranging from 45^{o}S to 45^{o}N requires only 26024 3D GCEs with respect to 52128 GCEs for a whole globe and becomes more computationally affordable with current computing resources.

It is well known that a latitude-longitude grid system has issues such as efficiency/performance and convergence problems near the poles. As the meta grid system in the mgGCM is no longer bound to the fvGCM's grid system, this meta-grid concept could help avoid the performance issues by implementing a quasi-uniform grid system (such as a cube grid or geodesic grid) into the mgGCE. Such a deployment should lead to a substantial performance increase since 95% of the computing time for the MMF is spent on the mgGCE.

The fundamental communication paradigm for data redistribution in this implementation is similar to the MLP [21], which was previously used for parallelization in single-component models with tremendous benefits. The methodology is extended here to a multi-component modeling system, showing an alternative and easy way for coupling multiple components. Further improvements in the implementation include an adoption of a more portable communication paradigm (such as MPI-1 or MPI-2) and/or a sophisticated modeling framework. While the current implementation in process management, data communication/redistribution, and synchronization is solely done with Unix system calls, earlier experiences with the parallelism implementation in the fvGCM have proven that this can be easily extended with an MPI-2 implementation [12]. A survey on existing frameworks such as Earth System Modeling Framework, (ESMF, http://www.esmf.ucar.edu/) or Partnership for Research Infrastructures in earth System Modeling (PRISM, http://www.prism.enes.org) is being conducted; however, it is too early to make a final

selection. First of all, no framework has yet demonstrated its superior scalability with a very large number of CPUs (e.g., thousands of CPUs) and secondly this MMF modeling system is so complex and "innovative" that it would take time for framework developers to include the MMF's requirements (e.g., a huge number of GCEs) in their frameworks.

Finally, as the clock speed of single-core CPUs is reaching the limits of physics, multi-core CPUs emerged with performance enhancement by adding additional cores into a socket. While multi-core CPUs have advantages such as lower power consumption and price/performance, the changes in CPU architectures have tremendous impact on software development and thereby on numerical models. Currently, two supercomputers with multi-core CPUs have been installed at NASA. They are called Discover with about 1,500 cores and Pleiades with 51,200 cores, respectively. A plan to take full advantage of the multi-core systems with the MMF is being proposed. Currently, promising performance with the MPI-OpenMP fvGCM on the Discover has been obtained (see Appendix A for details.)

Acknowledgements: We thank Dr. D. Anderson, Mr. S. Smith, and Mr. Joe Bredekamp for their support for developing and improving the MMF under the NASA Cloud Modeling and Analysis Initiative (CMAI), Advanced Information Systems Technology (AIST), and Applied Information Systems Research (AISR) program, respectively. The GCE is mainly supported by the NASA Atmospheric Dynamics and Thermodynamics Program and Tropical Rainfall Measuring Mission. Finally, we'd like to express our sincere thank to Dr. T. Lee, the NASA Advanced Supercomputing Division and NASA Center for Computational Sciences divisions for their strong support and use of computing and storage resources.

6. References

[1] Tao, W.-K., D. Anderson, R. Atlas, J. Chern, P. Houser, A. Hou, S. Lang, W. Lau, C. Peters-Lidard, R. Kakar, S. Kumar, W. Lapenta, X. Li, T. Matsui, R. Rienecker, B.-W. Shen, J. J. Shi, J. Simpson, and X. Zeng, 2008: A Goddard Multi-Scale Modeling System with Unified Physics. WCRP/GEWEX Newsletter, Vol 18, No 1, 6-8.

[2] R. Kerr, 2006: A Tempestuous Birth for Hurricane Climatology. *Science*, Vol **312**, 676-678.

[3] L., K. Bengtsson, I. Hodges, and M. Esch, 2007: Tropical cyclones in a T159 resolution global climate model: comparison with observations and re-analyses. Tellus A 59 (4), 396.416 doi:10.1111/j.1600-0870.2007.00236.x

[4] D. Randall, M. Khairoutdinov, A. Arakawa, W. Grabowski, 2003b: Breaking the Cloud Parameterization Deadlock. *Bull. Amer. Meteor. Soc.*, 1547-1564.

[5] R. Biswas, M.J. Aftosmis, C. Kiris, and B.-W. Shen, 2007: Petascale Computing: Impact on Future NASA Missions. Petascale Computing: Architectures and Algorithms, 29-46 (D. Bader, ed.), Chapman and Hall / CRC Press, Boca Raton, FL.

[6] Atlas, R., O. Reale, B.-W. Shen, S.-J. Lin, J.-D. Chern, W. Putman, T. Lee, K.-S. Yeh, M. Bosilovich, and J. Radakovich, 2005: Hurricane forecasting with the high-resolution NASA finite volume general circulation model. *Geophys. Res. Lett.*, **32**, L03801, doi:10.1029/2004GL021513.

[7] B.-W. Shen, R. Atlas, J.-D. Chern, O. Reale, S.-J. Lin, T. Lee, and J. Chang, 2006a: The 0.125 degree finite-volume General Circulation Model on the NASA Columbia Supercomputer: Preliminary Simulations of Mesoscale Vortices. *Geophys. Res. Lett.*, **33**, L05801, doi:10.1029/2005GL024594.

[8] B.-W. Shen, R. Atlas, O. Reale, S.-J. Lin, J.-D. Chern, J. Chang, C. Henze, and J.-L. Li, 2006b: Hurricane Forecasts with a Global Mesoscale-Resolving Model: Preliminary Results with Hurricane Katrina (2005). *Geophys. Res. Lett.,* **33**, L13813, doi:10.1029/2006GL026143.

[9] B.-W. Shen, W.-K. Tao, R. Atlas, T. Lee, O. Reale, J.-D. Chern, S.-J. Lin, J. Chang, C. Henze, J.-L. Li, 2006c: Hurricane Forecasts with a Global Mesoscal Model on the NASA Columbia Supercomputer, AGU 2006 Fall Meeting, December 11-16, 2006.

[10] S. Lang, W.-K. Tao, J. Simpson and B. Ferrier, 2003: Modeling of convective-stratiform precipitation processes: Sensitivity to partitioning methods, *J. Appl. Meteor.* **42**, 505-527.

[11] W.-K. Tao, C.-L. Shie, R. Johnson, S. Braun, J. Simpson, and P. E. Ciesielski, 2003: Convective Systems over South China Sea: Cloud-Resolving Model Simulations. *J. Atmos. Sci.*, **60**, 2929-2956.

[12] W. Putman, S.-J. Lin, and B.-W. Shen, 2005: Cross-Platform Performance of a Portable Communication Module and the NASA Finite Volume General Circulation Model. *International Journal of High Performance Computing Applications*. **19**: 213-223.

[13] J.-M. Juang, W.-K. Tao, X. Zeng, C.-L. Shie, S. Lang, and J. Simpson, 2007: Parallelization of NASA Goddard Cloud Ensemble Model for Massively Parallel Computing. *Terrestrial, Atmospheric and Oceanic Sciences.* (in press)

[14] W.-K. Tao and J. Simpson, 1993: The Goddard Cumulus Ensemble Model. Part I: Model description. *Terrestrial, Atmospheric and Oceanic Sciences*, **4**, 19-54.

[15] W.-K. Tao, and J. Simpson, C.-H. Sui, B. Ferrier, S. Lang, J. Scala, M.-D. Chou, and K. Pickering, 1993: Heating, moisture and water budgets of tropical and mid-latitude squall lines: Comparisons and sensitivity to longwave radiation. *J. Atmos. Sci.*, **50**, 673-690.

[16] S.-J. Lin, 2004: A "vertically Lagrangian" finite-volume dynamical core for global models. *Mon. Wea. Rev.*, **132**, 2293-2307.

[17] J. T. Kiehl, J. Hack, G. Bonan, B. Boville, B. Briegleb, D. Williamson, P. Rasch, 1996: Description of the NCAR Community Climate Model (CCM3). NCAR Technical Note.

[18] Y. Dai, X. Zeng, R. E. Dickinsonh, I. Baker, G. B. Bonan, M. G. Bosilovich, A. S. Denning, P. A. Dirmeyer, P. R. Houser, G. Niu, K. W. Oleson, C. Adam Schlosserl, and Z.-L. Yang, 2003: The Common Land Model. *Bull. Amer. Meteor. Soc.*, **84**, 1013-1023.

[19] S.-J. Lin, B.-W. Shen, W. P. Putman, J.-D. Chern, 2003: Application of the high-resolution finite-volume NASA/NCAR Climate Model for Medium-Range Weather Prediction Experiments. EGS - AGU - EUG Joint Assembly, Nice, France, 6 - 11 April 2003.

[20] J. R. Taft, 2001: Achieving 60 gflop/s on the production cfd code overflow-mlp. *Parallel Computing*, 27(4):521-536.

[21] J.-L. F. Li, D. Waliser, C. Woods, J. Teixeira, J. Bacmeister, J. Chern, B.-W. Shen, A. Tompkins, W. K. Tao, and M. Köhler (2008), Comparisons of satellites liquid water estimates to ECMWF and GMAO analyses, 20th century IPCC AR4 climate simulations, and GCM simulations, *Geophys. Res. Lett.*, *35*, L19710, doi:10.1029/2008GL035427.

Appendix A: A Benchmark with the fvGCM on a Multicore Supercomputer

Discover supercomputer, a Linux Networx cluster with 388 nodes and 776 Xeon dual-core CPUs, came into operation at the NASA Goddard Space Flight Center in late 2006. This multi-core (4 cores per node) supercomputer was built to mainly support large-scale Earth Science modeling. In this section, the fvGCM performance on the Discover is evaluated with numerical weather prediction (NWP) experiments at a 0.5 degree resolution. We perform tests with a fixed number of MPI processes, and then with a fixed number of cores. In the former (Figure A1), the model with (30 nodes, 4 cores per node) can produce an 85% of the throughput obtained from the run with (60 nodes, 2 cores per node). In the latter (Figure A2), the hybrid run with (30 MPI, 4 OpenMP) processes gives a comparable throughout to the one with (60 MPI, 2 OpenMP) processes. These results suggest that the model is scaled quite well when the number of cores doubles. By further comparison, our benchmark indicates that the MPI-OpenMP hybrid mode has a better performance than the MPI-only mode on the Discover. A similar conclusion with the model at higher (1/4 degree) resolution is also obtained. For a practical NWP application with (60 MPI, 4 OpenMP) processes, the model at the 0.5 degree resolution can perform 1020 simulation days per wall-clock day.

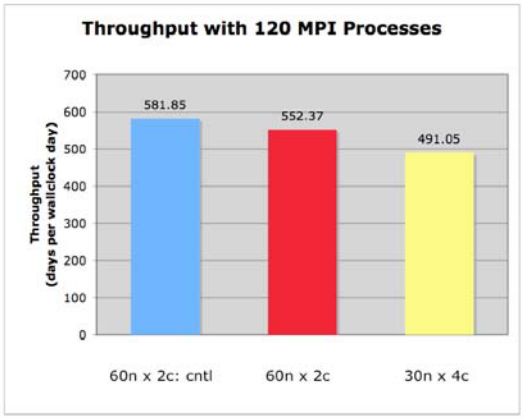

Figure A1: *Model performance with a fixed number of MPI processes. Number of (nodes, cores per node) shown by blue, red, yellow bars is (60, 2), (60, 2), and (30, 4), respectively. The term 'cntl' indicates the control run with default settings. The run with 30 nodes gives %85 of the throughput in the control run, but reduces %50 of the "cost" (i.e., # of nodes).*

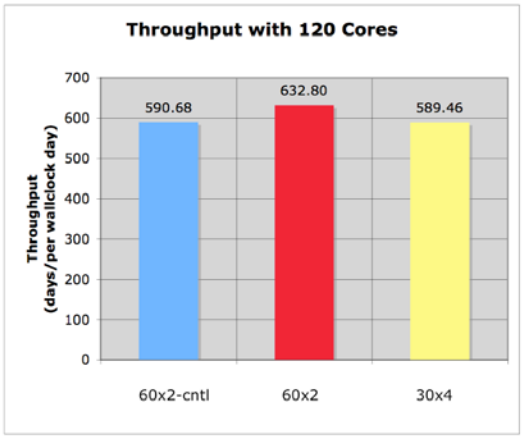

Figure A2: *Model performance with a fixed number of cores. Number of processes in (MPI, OpenMP) shown by blue, red, yellow bars is (60, 2), (60, 2), and (30, 4), respectively. The term 'cntl' indicates the control run with default settings. Compared to the control run labeled by the blue bar, the run with 30 nodes gives a comparable throughput with only half of the "cost" (i.e., # of nodes).*