Scaling climate simulation applications on the IBM Blue Gene/L system

J. M. Dennis
H. M. Tufo

We examine the ability of the IBM Blue Gene/L™ (BG/L) architecture to provide ultrahigh-resolution climate simulation capability. Our investigations show that it is possible to scale climate models to more than 32,000 processors on a 20-rack BG/L system using a variety of commonly employed techniques. One novel contribution is our load-balancing strategy that is based on newly developed space-filling curve partitioning algorithms. Here, we examine three models: the Parallel Ocean Program (POP), the Community Ice CodE (CICE), and the High-Order Method Modeling Environment (HOMME). The POP and CICE models are components of the next-generation Community Climate System Model (CCSM), which is based at the National Center for Atmospheric Research and is one of the leading coupled climate system models. HOMME is an experimental dynamical "core" (i.e., the CCSM component that calculates atmosphere dynamics) currently being evaluated within the Community Atmospheric Model, the atmospheric component of CCSM. For our scaling studies, we concentrate on 1/10° resolution simulations for CICE and POP, and 1/3° resolution for HOMME. The ability to simulate high resolutions on the massively parallel systems, which will dominate high-performance computing for the foreseeable future, is essential to the advancement of climate science.

1. Introduction and motivation

Enormous computational power is required to accurately simulate historical climate behavior or predict future climates. In the coming decade, one grand challenge that is being driven by increasingly likely global warming [1] is the need to understand the impact of various anthropogenic emission and atmospheric CO_2 concentration scenarios on the climate of the earth. The current generation of coupled climate system models can simulate most of the continental-scale features of the observed climate. However, these climate models still cannot simulate and, therefore, cannot project climate changes with the level of regional spatial accuracy desired for a more complete understanding of the detailed causes, effects, and impacts of climate change. We believe that the next generation of models requires much greater spatial resolution, inclusion of the full carbon, nitrogen, and biogeochemical cycles, as well as more-complete representations of physical processes (e.g., those that affect clouds, aerosols, ice sheets, land cover, and ocean mixing) in order to produce regionally improved simulations of climate.

Here, we examine the ability of the IBM Blue Gene* family of supercomputing systems to provide the desired increase in spatial resolution through benchmarking of the IBM Blue Gene/L* (BG/L) system, the first system within the family. Our objectives are to show that a significant leap forward in the resolutions at which climate is simulated is possible and to influence the design of future members of the Blue Gene family to ensure that they are capable of efficiently executing the next generation of climate models with extremely high processor counts. To this end, we concentrate on the scalability of three climate applications at ultrahigh spatial resolutions. The first two, the Parallel Ocean Program (POP) and the Community Ice CodE (CICE) application, are currently components of the next-generation Community Climate System Model (CCSM), one of the most extensively used climate models in the world. The third application involves the High-Order Method Modeling Environment (HOMME) that is

0018-8646/08/$5.00 © 2008 IBM

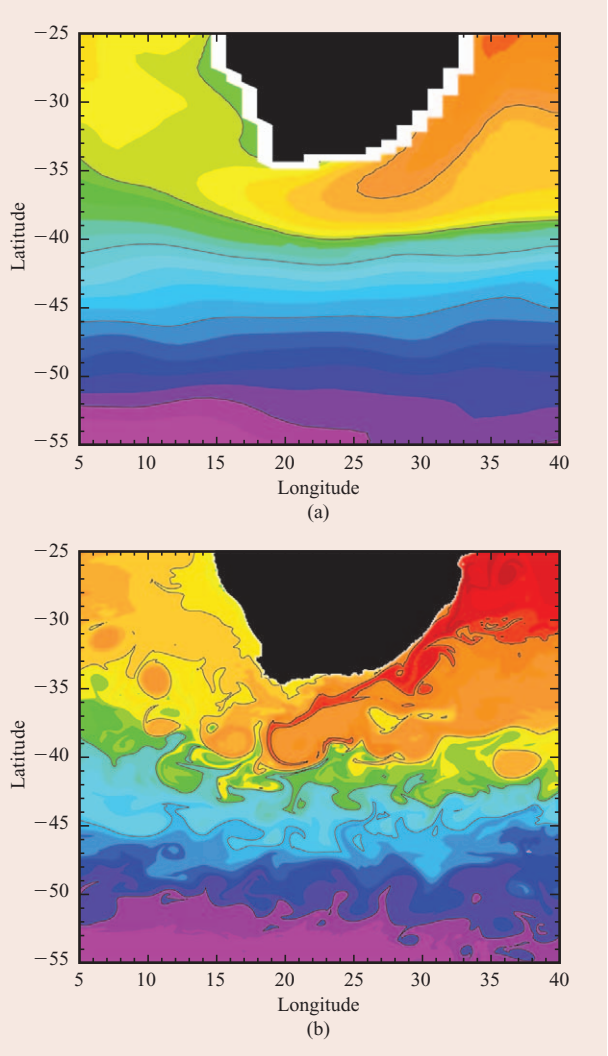

Figure 1

Sea surface temperature distribution in the region of the Southern Ocean surrounding the Cape of Good Hope as simulated by (a) the ocean component of the Community Climate System Model [2] and by (b) an eddy-resolving ocean general circulation model [3]. Axes are in units of degrees.

currently being evaluated within the Community Atmospheric Model (CAM), which is the atmospheric component of the CCSM. It should be emphasized that our HOMME results are more speculative in nature because HOMME is still under development. We are, therefore, only able to *estimate* the simulation rate of HOMME with realistic physics. We note that the CCSM also includes the Community Land Model that models the impact of the land surface on the climate, a coupler that addresses interactions between all component models, and other component models under evaluation such as those involving atmospheric chemistry, biogeochemistry, and ice-sheet dynamics. Given the scope of this work, we do not consider these additional CCSM components at present.

Our target ultrahigh spatial resolution corresponds to an average separation between grid points at the equator of 1/10° (i.e., 1/3,600 of the distance around the equator, or 11.1 km) for the ocean and sea ice models, and 1/3° for the atmospheric model. These resolutions are a compromise between the needs of climate scientists and what is computationally tractable within the next 5 years. Ideally, researchers would like to operate at 1-km "cloud clustering" resolutions (9/1,000°) for which it is believed that accurate simulation of critical convective processes is possible. However, a 1-km atmospheric model represents a 100-fold increase in resolution compared to that typically employed in CCSM at present. A 100-fold increase in resolution will result in a 1,000,000-fold increase in the computational cost with respect to the existing resolution used in CCSM. This increase would require greater than a sustained 100 petaflops to simulate 5 years per wall-clock day, which is considered the minimum threshold required to carry out climate studies. Unfortunately, no computing system capable of sustaining this flop (floating-point-operation) rate will be available within the next 5 years. In order to determine our target resolutions for this study, we held the integration rate constant and increased the resolutions for each model such that as many physical phenomena as possible are accurately represented. We anticipate that the future of massively parallel systems will enable the simulation of multiple ultrahigh-resolution simulations.

The impact of ultrahigh resolution is readily apparent within ocean models, and their relative simplicity with respect to atmospheric models makes them a better candidate for illustrating the impact (and limitations) of increased resolution. For example, a 1/10° resolution ocean model allows for a more realistic representation of ocean bottom bathymetry and coastal geometry, and it accounts for the majority of the ocean energy budget. A plot of the sea surface temperature distribution in the region of the Southern Ocean surrounding the Cape of Good Hope is provided in **Figure 1**. The various colors correspond to differences in sea surface temperature. Figure 1(a) represents a 1° simulation using the ocean component of the CCSM [2], while Figure 1(b) represents a 1/10° eddy-resolving ocean general circulation model [3]. The 1° simulation is incapable of capturing mesoscale eddies and, thus, their role in energy and carbon dioxide transport in the ocean.

In this paper, we demonstrate that it is indeed possible to use the BG/L system to perform ultrahigh-resolution climate simulations. In particular, we measure the simulation rate of the POP 1/10° benchmark on 28,972 BG/L processors to be 8.5 simulated years per wall-clock

day. An early version of CICE at 1/10° resolution achieves 40 simulated years per wall-clock day on 28,972 BG/L processors. HOMME achieves simulation rates of 19.7, 7.3, and 4.9 years per wall-clock day in an idealized test case on the BG/L system at 1/2°, 1/3°, and 1/4° resolutions, respectively. We note that an earlier version of HOMME, with an alternative physics formulation, demonstrates excellent scaling on 32,768 BG/L processors [4].

Achieving these integration rates with very high processor counts and at the target resolutions considered here requires employing several techniques to improve computation and communication load balance as well as eliminating all nonscalable data structures, algorithms, and implementations. In Section 2, we describe the application of an inverse space-filling curve (SFC) partitioning approach to the three models under consideration, and we discuss in detail how the weighted SFC (wSFC) variant is applied to the CICE model. Elimination of nonscalable constructs is described in Section 3. In Sections 4 and 5, we describe performance results obtained from running the CICE and POP models on a 20-rack BG/L system located at the IBM Thomas J. Watson Research Center. We observe that SFCs, in addition to other changes described in Section 3, double the simulation rate for the POP 1/10° benchmark at 28,972 processors, and the wSFC approach alone increases the simulation rate of an early version of CICE at 1/10° by 33% on 32,768 processors. In Section 6, we describe simulation rates of HOMME for an idealized test case at 1/2°, 1/3°, and 1/4° resolutions. Finally, in Section 7 we provide conclusions and discuss future work.

2. Partitioning the computational mesh for load balance

We describe how a computational mesh arising from the CICE model is partitioned across processors using SFCs, which are functions that map a higher to a lower dimensional space, in this case from a two-dimensional (2D) to a one-dimensional (1D) space (**Figure 2**). The use of SFCs within CICE is an extension of the partitioning technique initially developed for HOMME [5] and later applied to POP [6]. POP, CICE, and HOMME are hydrostatic models that solve the equations of motion on multiple coupled horizontal computational meshes. Without the use of the hydrostatic approximation, the equations and algorithms are considerably more complex. The size of the horizontal computational mesh that is decomposed across processors is significantly larger than the number of levels in the vertical dimension. (We refer to such levels as *vertical levels*.) For example, with a 1/10° POP, the horizontal dimension includes 3,600 × 2,400 grid points with 40 vertical levels. In the HOMME model, which supports both the spectral element and the

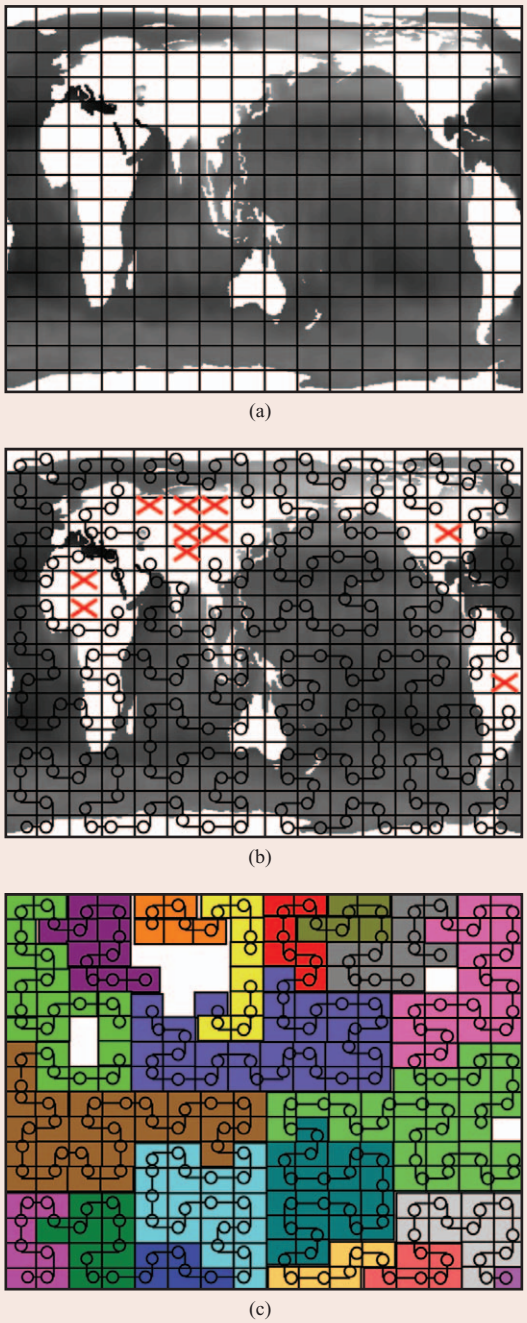

Figure 2

Computational mesh. (a) A global 1° CICE grid. Here, white corresponds to land points, gray to oceans points, and superimposed lines indicate blocks with 20 × 24 grid points [6]. (Republished with permission; ©2007 IEEE.) (b) A Hilbert (2^n) space-filling curve orders the blocks. Note that blocks corresponding to land, indicated by a red *X*, are eliminated from the computational domain. (c) A partitioning for 20 processors is indicated by the various colors. Note the larger domains at low latitude and smaller domains at high latitude.

discontinuous Galerkin [7–9] methods, the computational mesh is composed of a collection of elements with one or more elements being placed on each processor. Similarly, with the CICE and POP models, which share a common software infrastructure [10], the horizontal dimensions of the computational mesh are decomposed into logically rectangular 2D blocks. The computational mesh is distributed across multiple processors by placing one or more 2D blocks on each processor. A decomposition that allocates a single block per processor is hence referred to as a *single-block* configuration, compared to a *sub-block* configuration that allocates multiple blocks per processor.

The mapping of the computational grid into the BG/L 3D torus network occurs in two separate phases. In the first phase, blocks or elements are mapped into a 1D ordering using an SFC. The second phase consists of mapping the 1D ordering into the torus. We first describe the mapping of the computational domain into a 1D ordering.

We apply SFC partitioning to CICE by dividing the computational grid into 2D blocks such that the number of blocks in the x-direction (Nb_x) and y-direction (Nb_y) are $Nb_x = Nb_y = 2^n\ 3^m\ 5^p$ where n, m, and p are integers. The limit on the number of blocks in each horizontal dimension is due to a restriction on the types of SFCs that can be constructed. For more details on the construction of SFCs, see [6]. A global 1° CICE grid with 20×24 grid points per block is illustrated in Figure 2(a). The SFC that is applied to the CICE grid is illustrated in Figure 2(b). Note that blocks that correspond entirely to land points, which are indicated by a red X, are excluded from the curve. Partitioning across processors is achieved by subdividing the 1D curve into segments. Currently, for the POP and HOMME models, we assume that each block requires an equal amount of computational work. This assumption will not hold for HOMME when integrated into CAM and employed in simulations that contain a diurnal cycle because the radiation physics calculations induce a significant load imbalance. However, several techniques have been developed to address this diurnal load imbalance [11–13]. For the CICE model, this assumption does not hold, because the computational cost of blocks that contain sea ice is considerably higher than that of those that do not, and because a vast majority of the code within the CICE model is executed only if sea ice is present. Therefore, we apply a weight to every point in the SFC that corresponds to the computational work for each block. We estimate the weight (Wgt) for each block i to be

$$Wgt_i = w_0 + P_i w_1, \tag{1}$$

where w_0 is the amount of computational work for all non-land blocks, w_1 is the computational work for a block with sea ice, and P_i is the probability that a block i contains sea ice. We have examined several techniques in order to estimate the probability of sea ice within a block. We describe the impact of two such techniques: an error function ($erfc$)-based approach and a climatological approach. For the error function approach, we calculate the probability of sea ice

$$P_i = erfc([\varphi - \max(|lat_{ij}|)]/\sigma), \tag{2}$$

where lat_{ij} is the latitude for point j in block i, φ is the mean sea-ice extent, σ is the variance in sea-ice extent, and $erfc(\ )$ is the error function. We use a separate φ and σ for the northern and southern hemisphere. For the climatological approach,

$$P_i = \begin{cases} 1.0 & \text{if } \left(\sum_j \phi_{ij}/n_i\right) \geq 0.1 \\ 0.0 & \text{otherwise} \end{cases}, \tag{3}$$

where ϕ_{ij} is maximum sea-ice extent at point j within block i, and n_i is the number of points within block i with non-zero ϕ_{ij}. The value for ϕ_{ij} is derived from satellite observations of sea-ice concentrations. A "greedy" algorithm (an algorithm in which "customers" are served in order and each takes as much as he is able) is used to partition the wSFC such that the maximum amount of computational work on any single processor is minimized.

Once the 1D ordering of the computational grid is produced by the SFC, it is possible to map Message Passing Interface (MPI) processes into the BG/L torus. Several mappings have been used with the HOMME model [4], and they reveal that processor mapping becomes important for configurations with greater than about 12,000 processors. In particular, we have used the snake mapping, which uses a small 2D Morton SFC [14]. The default lexicographical ordering assigns tasks in 1×1 node blocks using XYZ ordering, in which X, Y, and Z refer to the dimensions (i.e., axes) of the torus. For example, the assignment of node (X,Y,Z) is followed by $(X+1,Y,Z)$. The snake ordering is also a lexicographical ordering but uses 2×2 node blocks. Therefore, assignment occurs within a 2×2 node block before incrementing the X, Y, or Z dimension. The snake mapping has better locality than the default lexicographical ordering and likely reduces message contention, which accounts for its superior performance at large processor counts. The mapping of MPI tasks to the BG/L torus is a potential application of forward SFC partitioning, i.e., the mapping of a lower-dimensional to a higher-dimensional space.

3. Eliminating nonscalable constructs

In addition to the SFC partitioning technique described in the previous section, several other considerations are

critical in the preparation of applications for the BG/L system. The most critical consideration is the elimination of all nonscalable data structures, algorithms, and implementations. Within both POP and CICE, an $O(N^2)$ algorithm is used to initialize the communication neighbors, where N is the number of blocks. We replace the $O(N^2)$ neighbor search algorithm with an $O(N)$ algorithm that stores a short neighbor list. The elimination of this algorithm reduces the time it takes for POP and CICE to initialize on large processor counts from 40 minutes to 40 seconds. The N^2 initialization algorithm is an example of a poorly designed algorithm that has likely existed within POP since its inception. Its poor design becomes evident only on large processor counts.

A problem common to both the POP and the CICE is limited parallelism within the disk I/O subsystem of the applications. For example, POP parallelizes only disk I/O across vertical levels, while the CICE I/O subsystem contains no parallelism. Within POP, this limits the number of possible MPI processes from which disk I/O is performed to 40. An even more critical flaw in the design is that it requires an all-to-one assembly of the distributed horizontal dimension. This assembly has the potential to exhaust the limited amount of memory on MPI processes performing the disk I/O for large processor configurations. Further, disk I/O on a single 2D horizontal variable is performed on a single processor. The existing design and the limited memory per processor of the BG/L system prevent a 1/10° resolution POP from writing a history file.

To address the lack of parallelism with the disk I/O subsystem, we have developed the parallel I/O (PIO) library. The PIO library is a high-level I/O library that uses MPI-IO and pNetCDF (network Common Data Form) [15] to provide parallel disk I/O. The Model Coupling Toolkit [16, 17] provides rearranger functionality. A *rearranger* moves data from one distributed decomposition to another. A rearranger is necessary because of limitations in the expressiveness of the pNetCDF library. Specifically, in order to achieve optimal performance from pNetCDF, the data of each MPI process must be rectangular in shape. This limitation of pNetCDF prevents the use of decompositions such as the one illustrated in Figure 2(c). While the PIO library increases the read and write bandwidths to disk, the crucial contribution of the library is its significant reduction of the maximum memory per MPI process in both the POP and the CICE model. The PIO interface has been implemented and tested in CAM, HOMME, and POP. The initial implementation of PIO within POP has increased disk bandwidth by about eightfold over serial I/O.

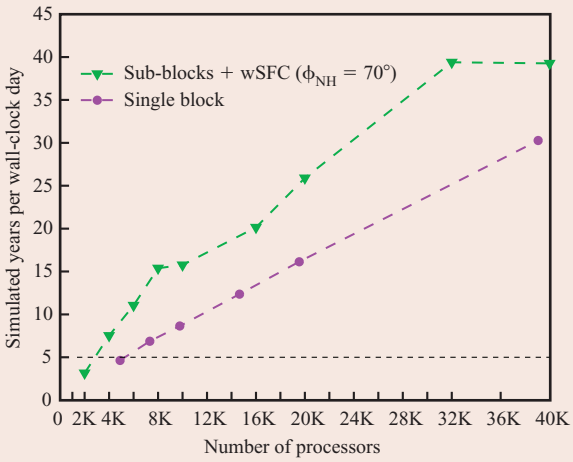

Figure 3

Plot of simulated years per wall-clock (elapsed) day for CICE at 1/10° on the BG/L system. (*x*-axis values indicate powers of 2, e.g., 2,048, 4,096, 8,192, ...; wSFC: weighted space-filling curve.)

4. Performance results for CICE

CICE decomposes the computational grid into 2D blocks that are distributed across processors. Traditionally, CICE has been configured such that a single block is allocated to each processor. (The block size is determined by dividing the computational grid into a 2D Cartesian grid.) This single-block configuration has the potential to create load imbalances that are due to the presence of land and the localization of sea ice to a small fraction of the computational grid. We examine the ability of wSFCs and the probability functions described in the previous section of this paper to reduce load imbalance.

With access to the BG/L system through the Consortium Days in which IBM offers time on the supercomputer to users [18], we examined the impact that different partitioning algorithms have on the simulation rate of CICE. We configured the CICE model for a 1-day initial run without atmospheric forcing at 1/10° resolution with a 30-minute timestep. We measured the simulation rate using both the single-block and the sub-block with wSFCs. Our BG/L results are based on the error function form of the probability estimation, where $\varphi_{NH} = 70°$ and $\sigma_{NH} = 5°$ for the northern hemisphere and $\varphi_{SH} = 60°$ and $\sigma_{SH} = 5°$ for the southern hemisphere. A plot of the simulation rates for CICE at 1/10° on the BG/L system is provided in **Figure 3**. The critical simulation rate of 5 years per wall-clock day is exceeded on 7,600 processors with the single-block configuration and on 4,096 processors for the wSFC configuration. Use of the wSFC partitioning doubles the simulation rate compared with the single-block configuration on 7,600 processors. The

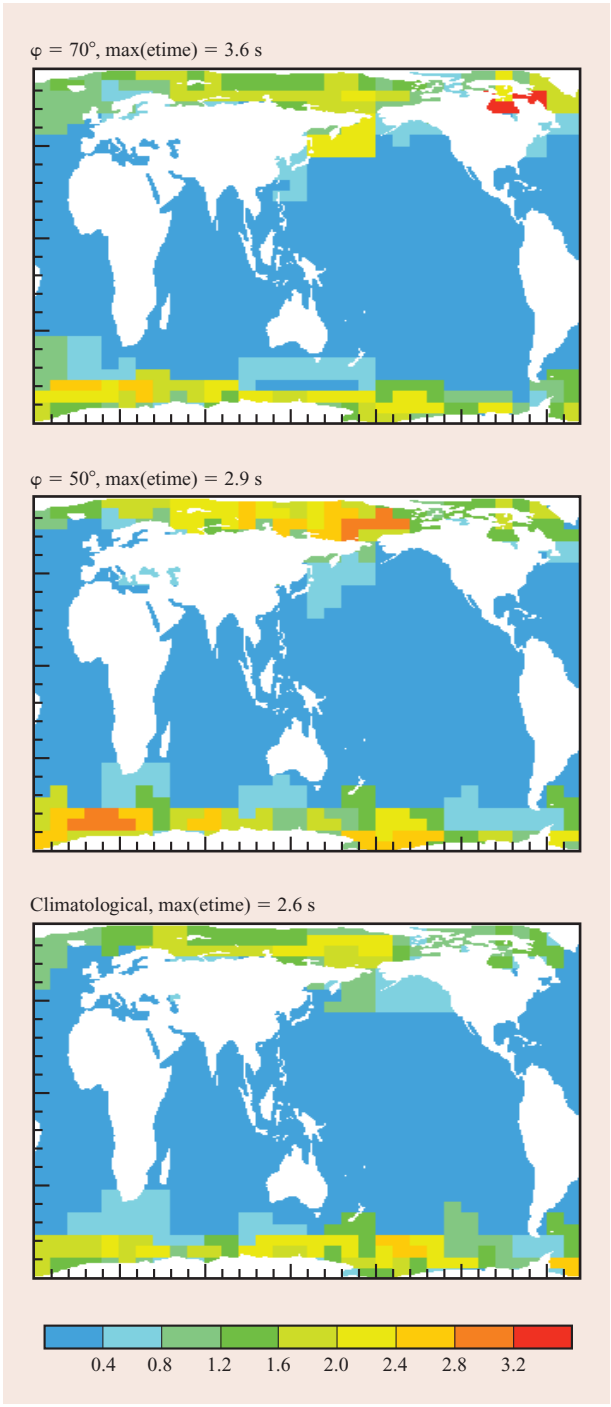

Figure 4

Spatial distribution of execution time in seconds for the dynamics sub-cycling component of CICE at 1° resolution on 160 BG/L processors. Top: Note the load imbalance (red area) due to sea ice in the Hudson Bay for erfc-based probability function for which $\varphi_{NH} = 70°$. Middle: The load imbalance is reduced when $\varphi_{NH} = 50°$. Bottom: The load imbalance is further reduced by using a climatology-based probability function. [Color indicates etime (execution time) in seconds.]

advantage of the wSFC-based partitioning decreases at larger processor counts. At 32,768 processors, wSFC-based partitioning provides only a 33% increase in simulation rate. The simulation rate for the wSFC configuration for processor counts of more than 7,600 suggests that we may not be accurately characterizing computational work.

We leverage the existing tools for visualizing physical fields to analyze possible sources of load imbalance not characterized by our function. We added software timers around the dynamics sub-cycling section of CICE that contains only those floating-point calculations that are due to the presence of sea ice. The time to execute the dynamics sub-cycling section of code is recorded to a full global 2D array and written to a history file. We executed CICE at 1° on 160 processors of the Blue Gene/L system at the National Center for Atmospheric Research (NCAR). **Figure 4** shows the spatial distribution of execution time in seconds for the dynamics sub-cycling for several different probability estimations. The top and middle panels in Figure 4 correspond to the error-based function approach in Equation (2), while the bottom panel is the climatological approach described in Equation (3). For the top panel of Figure 4, as alluded to previously, we used ($\varphi_{NH} = 70°$, $\sigma_{NH} = 5°$) for the Northern Hemisphere and ($\varphi_{SH} = 60°$, $\sigma_{SH} = 5°$) for the Southern Hemisphere. The red patch in this panel clearly indicates that the processor assigned to the Hudson Bay has a considerably longer execution time than any other processor. Our probability approximation that uses a Northern Hemisphere cutoff of $\varphi_{NH} = 70°$ does not account for the presence of sea ice in the Hudson Bay; the bay is considerably further south than the cutoff. The middle panel in Figure 4 shows the spatial distribution of execution time when the Northern Hemisphere cutoff is lowered to $\varphi_{NH} = 50°$. The error function probability estimate now accurately accounts for the presence of sea ice in the Hudson Bay, and the execution time is reduced to 2.9 seconds. However, the lowering of the Northern Hemisphere cutoff in order to address the presence of sea ice in the Hudson Bay overestimates the extent of sea ice in the Northern Hemisphere. A climatology-based approximation using Equation (2) provides an improved estimation of computational cost. The spatial distribution of execution time for a climatology-based approach is illustrated in the bottom panel of Figure 4. The execution time for the dynamics sub-cycling component of CICE is reduced to 2.6 seconds. While the use of wSFCs within the CICE model is preliminary, the potential decrease in execution time is promising.

5. Performance results for POP

As with CICE, POP decomposes the horizontal dimensions into 2D blocks that are distributed across

processors. Similarly, POP has been traditionally configured so that a single block, whose size is determined by dividing the horizontal computational grid into a 2D Cartesian grid, is allocated per processor. However, this configuration has the potential to create load imbalances because of the presence of land.

Researchers discovered that the presence of land in POP analyses increases the amount of data that must be loaded from the memory hierarchy to the CPU [19]. It is possible to eliminate the land within the barotropic component of POP, which is based on a preconditioned conjugate gradient solver, through the use of a 1D data structure. The use of the 1D data structure within the conjugate gradient solver reduced execution time for a 1° resolution POP simulation on 64 processors by 10%. The use of the 1D data structure also allows for a reduction in the volume of data passed between MPI processes.

In **Figure 5**, we provide a plot of the simulation rates for the POP 1/10° benchmark using the single-block configuration on the BG/L system. Note that the notations *1D-DS* and *2D-DS* in Figure 5 refer to the use of the 1D and 2D data structures within the conjugate gradient solver. The result shown in this figure for the single-block 2D-DS configuration on the BG/L system clearly illustrates scaling limitations. Use of the 1D-DS configuration improves scaling on the BG/L system, increasing the simulation rate from 3.9 to 6.2 simulated years per wall-clock day. The increased scalability is a direct result of increasing the single-processor performance and reducing the cost of communication. Scalability on the POP simulation on the BG/L system is further enhanced through the use of SFC partitioning, which reduces load imbalance. The threshold of 5 years per wall-clock day is exceeded on 14,486 processors with the use of SFC-based partitioning. The simulation rate on approximately 30K processors is increased by 118%, from 3.9 to 8.5 simulated years per wall-clock day compared with the base single-block 2D-DS configuration.

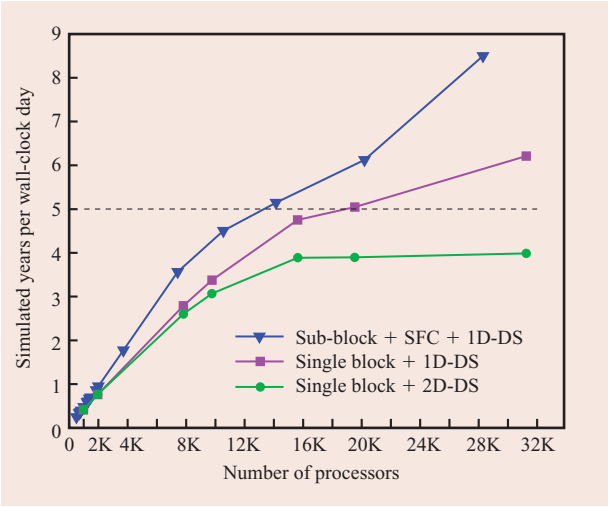

Figure 5

Plot of simulated years per wall-clock day for POP at 1/10° on the BG/L system. (DS: data structures; SFC: space-filling curve.)

6. Performance results for HOMME

The primary objective of the HOMME [20] initiative is the development of a class of high-order scalable conservative atmospheric models for climate and general atmospheric modeling applications. The spatial discretizations are derived from the spectral element (SE) and discontinuous Galerkin (DG) methods, which are local methods based on high-order accurate spectral basis functions that have been shown to perform well on massively parallel supercomputers at any resolution [7]. HOMME employs a cubed-sphere geometry exhibiting none of the singularities present in conventional latitude–longitude spherical geometries.

We examined the simulation rate for SE HOMME using explicit timestepping on the Held–Suarez test case [21]. The Held–Suarez test case is an idealized problem that is used to test the validity of an atmospheric dynamical core. The Held–Suarez test case does not include realistic physics that typically increases the cost of an atmospheric model by a factor of 2. Previous studies of the scalability of HOMME on the BG/L system [4] focused on the scalability of resolutions that matched the simulations performed with the AFES (atmospheric general circulation model for the Earth Simulator) [22] on the Earth Simulator [23]. Here, we concentrate on lower resolutions that would enable simulation rates of greater than 5 years per wall-clock day.

In particular, we examine three resolutions: 1/2°, 1/3°, and 1/4°. The cubed-sphere computational mesh used by HOMME is composed of six faces, where each face is composed of $Ne \times Ne$ block of elements. Each element contains $Np \times Np$ grid points, where the grid points on the boundary are shared with neighboring elements. (Ne is the number of elements on the face of the cube. Np refers to the number of pressure points within an element.) The average separation between grid points at the equator is $4 \times Ne \times (Np - 1)$, while the total number of elements is $6 \times Ne \times Ne$. We use both $Np = 4$ and $Np = 6$ for our study. For the 1/2° resolution case, we examine the scalability for $Ne = 36$, $Np = 6$ and $Ne = 60$, $Np = 4$. On the basis of our current 2D decomposition of the horizontal dimensions, which allocates one or more elements per processor, the two 1/2° configurations enable a maximum of 7,776- and 21,600-way parallelism, respectively. For the 1/3° resolution case, we use $Ne = 90$, $Np = 4$, which enables 48,600-way parallelism, and for the

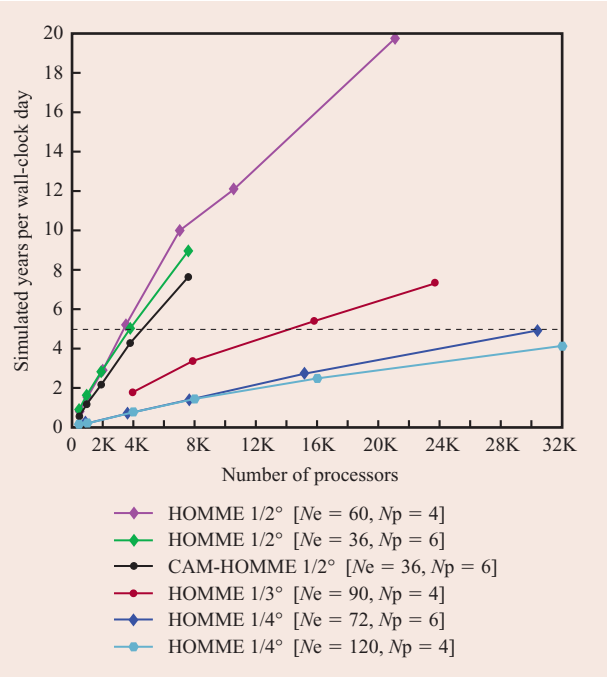

Figure 6

Plot of simulated years per wall-clock day for HOMME and CAM-HOMME on the Held–Suarez test case for 1/2°, 1/3°, and 1/4° resolution.

1/4° resolution case, we examine $Ne = 72$, $Np = 6$ and $Ne = 120$, $Np = 4$, which enables 31,104-way and 86,400-way parallelism, respectively. If the computational cost of the physics calculations becomes prohibitively expensive, through the addition of atmospheric chemistry or biogeochemistry, additional parallelism within HOMME is possible by decomposing the vertical dimension.

We provide a plot of the scalability for the 1/2°, 1/3°, and 1/4° resolution cases in **Figure 6**. Note that for the $Ne = 36$, $Np = 6$ case, we provide simulation rates for both HOMME and CAM-HOMME on 108 to 7,776 processors. *CAM-HOMME* refers to the version of CAM based on the HOMME dynamical core. The difference in simulation rate between HOMME and CAM-HOMME for the 1/2° case results from the inclusion of the physics–dynamics interface that converts between the data structures within the dynamics and physics components of CAM. Figure 6 clearly illustrates that HOMME is able to simulate the Held–Suarez test case at 1/2° at 8.9 years per wall-clock day on 7,776 processors for the $Ne = 36$, $Np = 6$ configuration and 19.7 years per wall-clock day on 21,600 processors for the $Ne = 60$, $Np = 4$ configuration. For the 1/3° resolution case, 7.3 years per wall-clock day is achieved on 24,300 processors. Finally, for the 1/4° resolution case, we achieve 4.9 years per wall-clock day on 31,104 processors for the $Ne = 72$, $Np = 6$ configuration, and 4.1 years per wall-clock day on 32,768 processors for the $Ne = 120$, $Np = 4$ configuration.

On the basis of the Held–Suarez results in Figure 6, we estimate that it is likely that a simulation rate of more than 5 years per wall-clock day is possible for CAM-HOMME with realistic physics for the 1/2° resolution case on 21,600 processors and for the 1/3° resolution case on 48,600 processors. It is unlikely, but possible, to sustain 5 years per wall-clock day at 1/4° on 86,400 processors. In the next section, we mention future modifications to the HOMME dynamical core that may increase the simulation rate.

7. Conclusions and future work

Access to the 20-rack Blue Gene/L system located at the IBM T. J. Watson Research Center allowed us to examine the scalability of three climate applications at ultrahigh spatial resolution. We discovered that it is possible to utilize 32K BG/L processors to achieve climatologically valid simulation rates for the POP and CICE models at 1/10°. Further, on the basis of our simulation rates for the Held–Suarez test case, we estimate that it should be possible to achieve a simulation rate in excess of 5 years per wall-clock day on both 1/2° resolution on 21,600 processors and 1/3° resolution on 48,600 processors. Our results clearly demonstrate the feasibility of utilizing large-scale parallelism to enable ultrahigh-resolution climate simulations.

Work is underway to prepare the remaining CCSM component models for efficient execution on the BG/L system. Reworking the initialization of the Community Land Model (CLM) has enabled an ultrahigh-resolution land model to be tested on the BG/L system. Within the CLM, a large amount of memory is consumed at initialization within a serial load-balancing algorithm. Use of a distributed load-balancing algorithm significantly reduces the memory footprint, enabling execution on the BG/L system while reducing the simulation rate by only 2%–3%. The reduction in the memory footprint of the initial implementation of the CLM and the reworked implementation is as large as about 50 times on 2,048 processors.[1] The existing *concurrent* coupler, which allows execution of each component on a separate set of processors, is replete with nonscalable memory usage, unnecessary communication, and load imbalances.[2] The existing concurrent coupler would require a complete rewrite to enable execution at high resolution on the BG/L system. As an alternative, a *sequential* coupler in which all processors execute all

[1]T. Craig, National Center for Atmospheric Research, Boulder, Colorado, personal communication.
[2]R. Jacob, Argonne National Laboratory, Argonne, Illinois, personal communication.

component models is being developed. The sequential coupler is designed and written with attention to minimizing the memory footprint and communication costs at high processor counts.[3]

There are multiple avenues for further development within each of the three component models. The POP model has a barotropic component that uses a preconditioned conjugate gradient algorithm to update the 2D surface pressure. It currently consumes 21% of the total time on 28,972 processors. We believe that the iteration count for the conjugate gradient solver could be reduced by 40% with the use of a sparse approximate inverse [24].

While the initial load-balancing results with the CICE model suggest a promising technique for reducing execution time, more work is needed. The greedy algorithm used to partition the wSFC does not take into account the communication cost of the partitioning. Our partitioning algorithm potentially creates unnecessarily large domains in which sea ice is not present. These large domains increase the communication cost for some large processor configurations. Our climatology-based probability function requires testing and refinement at the higher 1/10° resolution.

The HOMME model provides numerous avenues for additional work. The use of hyperviscosity within HOMME[4] may enable an increase in the length of the explicit timestep by 50%. The scalability of the semi-implicit timestepping version of HOMME is not fully understood. We have yet to validate and examine the performance of CAM-HOMME with realistic physics. Of particular importance is the need to address the diurnal load imbalance within the radiation calculation. One possible solution is to execute the dynamics component of CAM-HOMME on a subset of processors while CAM physics is executed on all processors. A wSFC partitioning based on the CICE work could be used to dynamically allocate the number of physics processors assigned to each dynamics processor.

Acknowledgments

The work of John Dennis is supported by the Department of Energy, CCPP Program grant no. DE-FC03-97ER62402, and that of Henry Tufo by the Department of Energy, SciDAC-CCPP grant no. DE-FG02-04ER63870. The National Science Foundation Cooperative Grant NSF01, which funds the National Center for Atmospheric Research (NCAR), supports both researchers. We thank Theron Voran, Ram Nair, and Mark Taylor for their help with HOMME. We thank Frank Bryan and Julie McClean for their collaboration on POP, and David Bailey, Elizabeth Hunke, and Bill Lipscomb for the collaboration on CICE. We also thank Ed Jedlicka of Argonne National Laboratory and Fred Mintzer of IBM Research for providing access to the 20-rack Blue Gene/L system through the second and fourth Blue Gene Watson Consortium Days event. Code development would not have been possible without access to the Blue Gene/L system operated by the NCAR, which is funded through the National Science Foundation MRI Grants CNS-0421498, CNS-0420873, and CNS-0420985, as well as through the IBM Shared University Research (SUR) Program with the University of Colorado.

[3]M. Vertenstein, National Center for Atmospheric Research, Boulder, Colorado, personal communication.
[4]M. Taylor, Sandia National Laboratory, Albuquerque, New Mexico, personal communication.

References

1. The Intergovernmental Panel on Climate Change (IPCC),"Climate Change 2007: The Physical Science Basis," *Summary for Policymakers, Contribution of Working Group I to the Fourth Assessment Report of the Intergovernmental Panel on Climate Change*, 2007; see *http://www.aaas.org/news/press_room/climate_change/media/4th_spm2feb07.pdf*.
2. W. D. Collins, C. M. Bitz, M. L. Blackmon, G. B. Bonan, C. S. Bretherton, J. A. Carton, P. Chang, et al., "The Community Climate System Model Version 3 (CCSM3)," *J. Climate* **19**, No. 11, 2122–2143 (2006).
3. M. E. Maltrud and J. L. McClean, "An Eddy Resolving Global 1/10° Ocean Simulation," *Ocean Modeling* **8**, No. 1-2, 31–54 (2005).
4. G. Almasi, G. Bhanot, D. Chen, M. Eleftheriou, B. Fitch, A. Gara, R. Germain, et al., "Early Experience with Scientific Applications on the Blue Gene/L Supercomputer," *Euro-Par 2005 Parallel Processing*, Vol. 3648, 2005, pp. 560–570; see *http://www.springerlink.com/content/7xl8hhbhbacp0vav/fulltext.pdf*.
5. J. M. Dennis, "Partitioning with Space-Filling Curves on the Cubed-Sphere," *Proceedings of the 17th International Symposium on Parallel and Distributed Processing*, 2003.
6. J. M. Dennis, "Inverse Space-Filling Curve Partitioning of a Global Ocean Model," *IEEE International Parallel and Distributed Processing Symposium*, 2007, pp. 1–10; see *http://www.cecs.uci.edu/~papers/ipdps07/pdfs/IPDPS-1569010963-paper-2.pdf*.
7. R. D. Loft, S. J. Thomas, and J. M. Dennis, "Terascale Spectral Element Dynamical Core for Atmospheric General Circulation Models," *Proceedings of the 2001 ACM/IEEE Conference on Supercomputing*, 2001; see *http://delivery.acm.org/10.1145/590000/582052/p18-loft.pdf?key1=582052&key2=0356663911&coll=GUIDE&dl=GUIDE&CFID=4568951&CFTOKEN=48073844*.
8. R. D. Nair, S. J. Thomas, and R. D. Loft, "A Discontinuous Galerkin Transport Scheme on the Cubed Sphere," *Monthly Weather Review* **133**, 814–828 (2005).
9. J. M. Dennis, R. D. Nair, H. M. Tufo, M. Levy, and T. Voran, "Development of a Scalable Global Discontinuous Galerkin Atmospheric Model," 2006; see *http://www.csc.cs.colorado.edu/~tufo/pubs/tufo-2005-ijcse.pdf*.
10. P. W. Jones, P. H. Worley, Y. Yoshida, J. B. White III, and J. Levesque, "Practical Performance Portability in the Parallel Ocean Program (POP): Research Articles," *Concurr. Comput. Pract. Exper.* **17**, No. 10, 1317–1327 (2005).

11. I. T. Foster and B. R. Toonen, "Load-Balancing Algorithms for Climate Models," *Proceedings of the Scalable High-Performance Computing Conference*, 1994, pp. 674–681.
12. S. P. Muszala, G. Alaghband, D. A. Connors, and J. J. Hack, "A VFSA Scheduler for Radiative Transfer Data in Climate Models," *Proceedings of the 17th International Conference on Parallel and Distributed Computing*, 2004; see *http://ece.colorado.edu/~muszala/pubs/54.pdf*.
13. S. P. Muszala, J. J. Hack, D. A. Connors, and G. Alaghband, "The Promise of Load-Balancing the Parameterization of Moist Convection Using a Model Data Load Index." *J. Atmos. Oceanic Technol. Am. Meteorol. Soc.* **23**, No. 4, 525–537 (2006).
14. H. Sagan, *Space-Filling Curves*, Springer-Verlag, New York, 1994.
15. J. Li, W.-K. Liao, A. Choudhary, R. Ross, R. Thakur, W. Gropp, R. Latham, A. Siegel, B. Gallagher, and M. Zingale, "Parallel netCDF: A High-Performance Scientific I/O Interface," *Proceedings of the 2003 ACM/IEEE Conference on Supercomputing*, 2003.
16. J. Larson, R. Jacob, and E. Ong, "The Model Coupling Toolkit: A New Fortran90 Toolkit for Building Multiphysics Parallel Coupled Models," *Int. J. High Perf. Comput. Appl.* **19**, No. 3, 277–292 (2005).
17. R. Jacob, J. Larson, and E. Ong, "MxN Communication and Parallel Interpolation in CCSM3 Using the Model Coupling Toolkit," *Int. J. High Perf. Comput. Appl.* **19**, No. 3, 293–307 (2005).
18. Blue Gene Consortium; see *http://www-fp.mcs.anl.gov/bgl2/*.
19. J. M. Dennis and E. R. Jessup, "Applying Automated Memory Analysis to Improve Iterative Algorithms," *SIAM J. Sci. Comput.* **29**, No. 5, 2210–2223 (2007).
20. HOMME on the IBM Blue Gene/L; see *http://www.homme.ucar.edu*.
21. I. M. Held and M. J. Suarez, "A Proposal for the Intercomparison of the Dynamical Cores of Atmospheric General Circulation Models," *Bull. Am. Meteorol. Soc.* **75**, No. 10, 1825–1830 (1994).
22. S. Shingu, H. Takahara, H. Fuchigami, M. Yamada, Y. Tsuda, W. Ohfuchi, Y. Sasaki, et al., "A 26.58 Tflops Global Atmospheric Simulation with the Spectral Transform Method on the Earth Simulator," *Proceedings of the ACM/IEEE Supercomputing Conference*, 2002; see *http://delivery.acm.org/10.1145/770000/762828/p67-shingu.pdf?key1=762828&key2=9079663911&coll=GUIDE&dl=GUIDE&CFID=4573887&CFTOKEN=92634490*.
23. S. Habata, M. Yokokawa, and S. Kitawaki, "The Earth Simulator," *NEC Res. Dev.* **44**, No. 1, 21–26 (2003).
24. M. J. Grote and T. Huckle, "Parallel Preconditioning with Sparse Approximate Inverses," *SIAM J. Sci. Comput.* **18**, No. 3, 838–853 (1997).

Received March 17, 2007; accepted for publication April 7, 2007; Internet publication December 19, 2007

John M. Dennis *Computer Information and Systems Laboratory, National Center for Atmospheric Research, P.O. Box 3000, Boulder, Colorado 80307 (dennis@ncar.edu)*. Dr. Dennis received his Ph.D. degree in computer science from the University of Colorado at Boulder. He studied the memory efficiency of Krylov iterative algorithms. He is one of the Chief Architects and the Lead Developer of the HOMME framework. Dr. Dennis's work has been recognized with an honorable mention Gordon Bell Prize in 2001 for excellence in high-performance supercomputing. His interests include high-performance parallel computing, mesh partitioning, computer architecture, iterative algorithms, and language processing.

Henry M. Tufo *Computer Information and Systems Laboratory, National Center for Atmospheric Research, P.O. Box 3000, Boulder, Colorado 80307 (tufo@ncar.edu)*. Dr. Tufo received his Ph.D. degree from Brown University in 1998, and he is currently an Associate Professor of computer science at the University of Colorado and leads the Computer Science Section in the Computational and Information Systems Laboratory at the National Center for Atmospheric Research. Dr. Tufo conducts research in high-performance computing, parallel algorithms and architectures, high-order numerical methods, scalable solvers, computational fluid dynamics, atmospheric science, scientific visualization, Linux** cluster technology, and grid computing. Dr. Tufo's work has been recognized with Gordon Bell Prizes in 1999 and 2000 for demonstrated excellence in high-performance supercomputing.