

Using spatial principles to optimize distributed computing for enabling the physical science discoveries

Chaowei Yang¹, Huayi Wu, Qunying Huang, Zhenlong Li, and Jing Li

Joint Center for Intelligent Spatial Computing, College of Science, George Mason University, Fairfax, VA 22030-4444

Edited by Michael Goodchild, University of California, Santa Barbara, CA, and approved December 14, 2010 (received for review August 15, 2009)

Contemporary physical science studies rely on the effective analyses of geographically dispersed spatial data and simulations of physical phenomena. Single computers and generic high-end computing are not sufficient to process the data for complex physical science analysis and simulations, which can be successfully supported only through distributed computing, best optimized through the application of spatial principles. Spatial computing, the computing aspect of a spatial cyberinfrastructure, refers to a computing paradigm that utilizes spatial principles to optimize distributed computers to catalyze advancements in the physical sciences. Spatial principles govern the interactions between scientific parameters across space and time by providing the spatial connections and constraints to drive the progression of the phenomena. Therefore, spatial computing studies could better position us to leverage spatial principles in simulating physical phenomena and, by extension, advance the physical sciences. Using geospatial science as an example, this paper illustrates through three research examples how spatial computing could (i) enable data intensive science with efficient data/services search, access, and utilization, (ii) facilitate physical science studies with enabling high-performance computing capabilities, and (iii) empower scientists with multidimensional visualization tools to understand observations and simulations. The research examples demonstrate that spatial computing is of critical importance to design computing methods to catalyze physical science studies with better data access, phenomena simulation, and analytical visualization. We envision that spatial computing will become a core technology that drives fundamental physical science advancements in the 21st century.

Globalization over the past decades has raised human awareness of global challenges, such as global warming, severe weather conditions and rapid diffusion of contagious diseases (1, 2). These challenges require advancement in the physical sciences to better understand the world around us. Spatial data, the information related to space and time, are critical to scientific advancement by providing multidimensional (3D space, 1D time, and important parameters) records of the Earth. Large-scale data are typically collected, generated, and stored in geographically dispersed locations and must therefore be supported by distributed computing facilities. Seamless sharing and access to these resources calls for a spatial cyberinfrastructure (CI) to enable the use of spatial data for the advancement of the physical sciences (1, 3, 4). Global initiatives, such as the Global Earth Observation System of Systems (GEOSS, an initiative to integrate Earth observation data to address regional to global problems such as climate change), Digital Earth (an initiative to integrate Earth referenced data to provide intuitive and better end-user tools), and the Spatial Data Infrastructure (SDI, an initiative to build an infrastructure to seamlessly share spatial data) (5–8), all call for the integration of many spatial resources to solve complex problems, such as climate change prediction and rapid emergency response. These problems must be addressed in a multidimensional context and conceptualized as spatiotemporal or geospatial dynamics in order to be effectively resolved (9, 10).

Because of the distributed nature of physical science resources and computing devices, a computing paradigm utilizing intrinsic spatial principles is needed to optimize distributed resources

(10–12). We refer to this computing paradigm as spatial computing. Spatial computing leverages spatial principles, such as space and time connections and constraints (13), in computing arrangements, selection, and use of data to make possible the computability of physical science problems. For example, spatial computing can be used for (i) understanding dynamic domain decomposition, where different resolutions need different decompositions of domains interactively within a simulation process, such as dust-storm forecasting (14, 15) and routing (16), in a self-adaptive fashion (17); (ii) synchronization for dynamic simulations to remove the errors introduced by simulations in each step through the decoupled domain borders; (iii) visualizing datasets using spatial computing in a distributed environment; and (iv) building spatial CI by providing spatial computing methodologies for both transparent and opaque cloud computing (4, 18–20).

Here, we consider spatial principles as the spatial and temporal connections and constraints among phenomena (or scientific parameters). The principles include (13, 17, 21–23)

1. Physical phenomena are continuous and digital representations (scientific parameters) are discrete for both space and time
 - a. Closer things are more related than those farther away:
 - i. Correlations exist among parameters, time, and space
 - ii. Neighboring discrete representation cells need communication across time
 - iii. Duplication along domain borders is needed for analyzing/simulating phenomena
 - b. Multiscalar
 - i. Phenomena are represented at global, regional, and local scales
 - ii. Human thinking and digital representation of phenomena are hierarchical and fractal
 - iii. Information frequency determines the hierarchical structure
2. Physical phenomena are heterogeneous in space and time
 - a. Higher resolution will include more information
 - b. Phenomena evolve at different speeds (the faster a dynamic process, the quicker an exchange occurs among neighbors)
 - c. The longer a dynamic process persists and the larger its spatial scale, the more exchanges are needed among neighbors

These general spatial principles can help guide us in designing solutions for spatial CI to enable greater physical science discoveries. National Science Foundation denoted CI with four major aspects including data access, high-performance computing, visualization, and education (24). Taking dust-storm research as an example, we illustrate how spatial computing can be used to help construct a spatial CI to solve physical science problems associated with data access, high-performance computing, and visualization. We discuss utilizing these spatial principles in three examples from the aspects of (i) the relevant physical science problems, (ii) experi-

Author contributions: C.Y. designed research; H.W. managed research; and C.Y., H.W., Q.H., Z.L., and J.L. performed research, analyzed data, and wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

¹To whom correspondence should be addressed. E-mail: cyang3@gmu.edu.

ments for finding/identifying spatial principles, (iii) utilization of spatial principles in our experiments, and (iv) enablement for solving these problems and applicability of spatial principles for solving other related problems. All three research examples are focused on geospatial phenomena. Other spatial problems, such as those related to topology and hypercube, can also be tackled with a spatial CI because they share similar characteristics.

Data/Service Search, Access, and Utilization for Climate Change Analysis

Global climate change has generated more frequent and intense severe weather patterns in recent years. It is a great challenge for the Earth sciences to identify and characterize the physical parameters contributing to the formation of severe weather. For example, to tease out the driving forces of a dust storm, we must access and analyze over 100 dynamic or static physical parameters recorded as spatial data including relative humidity, temperature, U wind, and V wind. These dynamic data reside at distributed locations, such as National Aeronautics and Space Administration (NASA) and National Oceanic and Atmospheric Administration centers, and are provided through standard interfaces (14, 23), such as a Web Coverage Service or a Web Map Service (WMS). For end users, traditional data access methods require ordering, data copying, shipping, and loading, a process that could last from days to months. Effective integration and analysis of these distributed data through Web services can help us conduct research more effectively by providing near real-time online data access (25).

A spatial CI should integrate these services for users with acceptable quality and performance levels when services accessed have varied performance levels (a common reality). Thousands of WMSs are freely available online, but their discovery is an arduous task for end users. In addition, access to WMSs requires the retrieval of a detailed capability file (including a description of the content, extent, and resolution of the available data products), which is time-consuming to download and process. The response time for accessing a WMS ranges from milliseconds to hours. Long response times are not tolerable in the Internet arena, especially when responding to emergency situations (26). These services are normally accessed by spatial Web portals or geobrowsers (27, 28), where processing occurs at the portal server. There are also performance problems when accessing data from multiple services with varying performance levels. Our research explores the possibility of applying spatial principles to design and achieve better performance from spatial Web services for scientists and other users across distributed geographic locations.

We picked 15 WMSs, evenly distributed across North America with a varying number of datasets served, and tested their performance by accessing them from three distributed sites. To ensure results' comparability, the sites have 100-Mbps connection to a campus Internet backbone with similar hardware/software configurations. Fig. 1*A* reveals *spatial principle 1.a* showing that the response time is related to distance, in either a physical or a virtual environment between the client and the service server. For example, the Fairfax site performs much better than the other two sites when accessing servers that are close to Fairfax. This locality effect is of critical importance when deploying an operational spatial CI to provide sufficient performance for distributed users.

To address the performance problems of capability file downloading and preprocessing, a centralized server resides at the same location as the portal server and can be employed to harvest and broker the capability file information. Thus, a portal server can access capability information in a centralized, rather than in a distributed, manner. Fig. 1*B* illustrates capability information access performance with and without using the centralized server from the three sites. The bars shown are performance averages across time based on the three sites' performance. Because each layer of a WMS has its own quality and performance levels (24, 26), we introduced a layer-based Application Programming

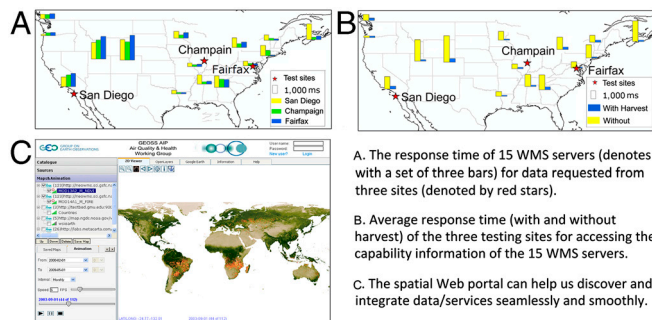


Fig. 1. Performance and quality of Web-based services are spatially collocated and essential to provide adequate support to data selection in scientific research.

Interface (API) search so that the performance is more predictable. This configuration provides a higher performance access model for published WMS layers. However, the introduction of the API incurs costs to the spatial interoperability of a WMS.

Another issue in searching, accessing, and utilizing distributed services is quality of service (QoS), including metrics such as availability, performance, and stability. Although the consumer cannot control QoS, providing QoS information in a spatial Web portal can help users identify better services. An initial QoS function is defined as Eq. 1 to evaluate the server performance, including the time to (i) connect to the server, (ii) process the request on the server side, and (iii) download the response from the server. Eq. 1 represents the server performance (ST) by subtracting the download speed, denoted by volume divided by Internet speed (S/V), from the total response time (T).

$$ST = T - S/V, \quad [1]$$

where T is in milliseconds, S is in bytes, and V is in bytes/millisecond.

To obtain a more comprehensive result, Eq. 2 was developed to calculate the average response time by repeating the test N times. The tests are distributed at multiple geographic locations and at different times according to the spatial principle 2.b:

$$\overline{ST} = \sum_{i=1}^N (T_i - S_i/V_i)/N, \quad [2]$$

where \overline{ST} = average server performance, N = number of test times (including different locations), and i denotes the i th time.

To eliminate the service time sensitivity, the average server performance is utilized in a spatial Web portal (see Fig. 1*C*) to classify service layer performance into six grades signaled by five green bars to support the usage by the GEOSS Air Quality and Health Working Group. The more green bars, as illustrated in Fig. 1*C*, the better the performance. The portal also supports multiple datasets (such as vegetation cover and forest fires) to be shown over time and space, to illustrate the correlation of physical science parameters (vegetation cover and fires).

Our portal experiment found that 200 WMS capability files can be accessed within 1 s (25). Therefore, we can discover and utilize datasets for scientific studies on the fly instead of waiting for days to months to get the datasets and determine if the datasets obtained are usable or not. This performance improvement contributes to a 21st century paradigm shift in data access—from media shipping to on the fly, online access. Our study illustrates how to identify spatial principles and utilize the identified principles to design a spatial Web portal, a critical interface for spatial CI, for higher performance. It is also straightforward to scale up the above architecture further for even more widely dispersed geographic locations, according to spatial principles 1.b and 2.b, by deploying several centralized servers (distributed for serving different regions).

This research advances the study of QoS and Web services for spatial CI to enable data access solutions for highly complex Earth science problems. These previously undescribed methodologies and tools can be used to identify and select the best services in a

timely fashion and deliver them into phenomena simulations. We are using these research results in deploying the GEOSS clearinghouse.

High-Performance Spatial Computing for Enabling Dust-Storm Research

Windblown dust is a primary natural hazard causing public health and environmental problems. To mitigate dust storms, atmospheric scientists try to determine the relevant parameters, such as soil and vegetation types, contributing to the generation of dust storms to improve dust-storm prediction (14). This research task poses significant challenges to computing facilities. For example, when conducting the initial research, we could simulate a geographic region only of about $2,000 \times 2,000 \text{ km}^2$ at $10 \times 10 \text{ km}^2$ resolution. The simulation model ran an entire day to produce a three-day result. This poor performance prevents researchers from achieving higher resolution, longer time, or broader geographic coverage in dust-storm simulation models. To enable higher resolution and broader coverage, researchers can tap into concurrency and high-performance computing (HPC) (28, 29) and develop spatial CI to tackle such computational intensive problems (2, 3, 30–33).

This section reports the results of five dust-storm modeling experiments to obtain finer resolutions, larger domains, and longer time scales by exploring, understanding, and utilizing spatial principles as they pertain to high-performance computing. We chose the Weather Research and Forecasting–Nonhydrostatic Mesoscale Model (WRF-NMM, <http://www.dtcenter.org/wrf-nmm/users>) and NMM-dust (34–36) for these experiments. These simulations include numerical weather models with appropriate boundary conditions (14, 37), where the physical space is usually decoupled in regular grid cells and the numerical computations on inner grid cells are performed uniformly (38). The computational cost of the models is a function of the number of cells, the time step, and algorithms used for the domain (39).

Parallelization Degree. Parallelization degree refers to how many central processing unit (CPU) cores can be leveraged in a concurrent mode effectively (40). Most parallel implementations of atmospheric models use the domain decomposition method in one form or the other, and the grid point models (e.g., WRF-NMM) require nearest-neighbor communication in the physical domain to keep consistent the progression of physical phenomena (41). Parallel implementations of dust-storm models indicate the spatial principle 1.a.ii “neighbor discrete representation cells need communication cross time.” The test uses the dust models for the southeastern United States to find out how many CPU cores can be utilized with a cluster of 28 computing nodes connected to each other with 1 Gbps. Each node in the cluster has two quad-core processors (eight physical cores) with a clock frequency of 2.33 GHz. We first parallelized the dust model and then tested the parallelized version on a cluster (14). The experiment demonstrates that the model can leverage only a limited number of CPU cores (up to 80 as illustrated in Fig. 2A).

We found that the dust model does not scale well beyond 48 cores and produces no further performance gains after 80 cores are utilized. This saturation results from the overhead for exchanging boundary information and synchronizing data above 80 cores. However, after reaching the best performance point (80 cores), there are still several minutes of fluctuation beyond 80 cores due to the trade-off between the increase in communication overhead and computing power.

Decomposition Method. The domain of dust-storm model (NMM-dust) is decomposed into multiple subdomains along longitude and latitude directions to achieve parallelization, and each processor computes one subdomain. However, dynamics are not consistent along the space, e.g., velocities are relatively large near the poles and are much smaller in the North–South (meridional direction) than that of in the East–West (zonal direction) (41). Spatial principle

2 “spatial heterogeneity of physical phenomena” can be found in the noneven dynamics characteristic of atmospheric circulation. Therefore, communication needs differ among processors in the South–North (S–N) direction from those of the West–East (W–E) direction (41). In addition, different domain sizes along W–E and S–N directions cause different numbers of grid cells along these two directions. This results in different amounts of grid cells to exchange boundary conditions along W–E and S–N directions. Thus, for the same degree of parallelization, different decompositions can result in a different communication overhead. We tested the parallel implementation through various decompositions of 24 subdomains along S–N and W–E directions from the same domain (Fig. 2B). We observed that a one-dimensional decomposition in both longitude and latitude alone is a bad idea for parallel implementation as the 24×1 (24 columns along S–N and only 1 column along W–E) had the worst performance followed by the 1×24 (1 columns along S–N and only 24 column along W–E), and more decomposition along longitude (S–N) direction is preferred as 3×8 and 4×6 decompositions obtain higher performance than that of 8×3 and 6×4 (Fig. 2C).

Spatial Resolution. Short-range dust-storm prediction is time sensitive (typically should be completed in $<2 \text{ h}$, 42) and computationally intensive. The time-sensitivity requirement is used as a criterion to identify the number of cores required for predictions as a function of spatial and temporal resolution. According to spatial principle 2.a, physical phenomena are spatially heterogeneous and a higher resolution will include more information. Therefore, more computations are required to simulate the phenomena at a higher resolution. This experiment was designed to analyze the relationship between the spatial resolutions of the dust-storm prediction model and the number of cores. Our results show that one core is sufficient for successfully completing the 10-km resolution simulation in 2 h (Fig. 2D). Eight CPU cores are required for the 5-km simulation, whereas at least 16 cores are needed for a 4-km simulation. Computation requirements are greatly increased with increased resolution. The computing time of 5-km resolution and 4-km resolution simulations increases by a factor of 10.5 and 16.5, respectively, when compared with the 10-km resolution simulation using only one core.

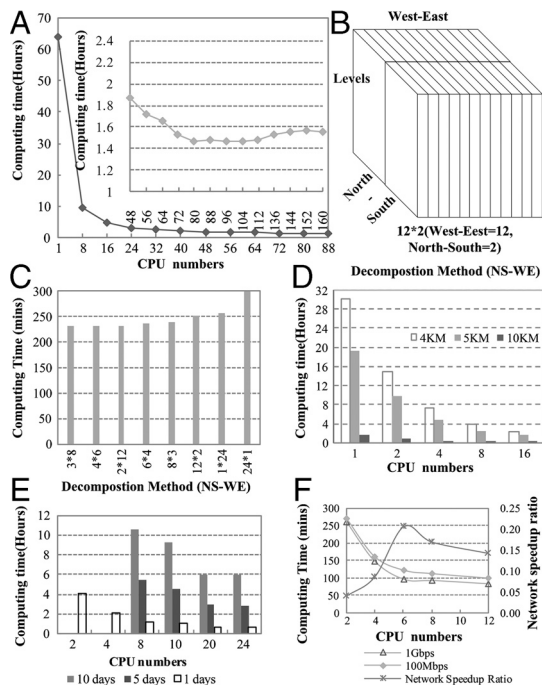


Fig. 2. Spatial principles are utilized to guide optimizing high performance computing to enable phenomena simulation for better resolution, time, and geographic scope.

Theoretically, the computational cost of an explicit three-dimensional hydrodynamics weather forecasting model behaves like a function of n^3 for a given domain size, where n is a grid dimension (38). Therefore, the theoretical increase in computing time from 10- to 5- and 4-km resolution should be about 8 in the former instance and 15.6 in the latter. The actual increases in computing time are higher than these theoretical estimates because of the additional communication overhead required. We found in this experiment that a resolution increase to 3 km in each dimension of the model cannot be performed in the cluster for the domain of 2,000 km \times 2,000 km \times 37 levels. To support the runs, we should either (i) redesign the existing algorithms, codes, and data structures or (ii) increase the speed of the CPU and the network connection. Thus this study poses a great challenge for both physical science (in improving model resolution and extent) and computer science (in increasing CPU and network speed).

Temporal Scope. Another forecasting capability is for a computing platform to support long-term dust-storm predictions (Fig. 2E). It is observed that 4 CPUs can successfully complete a one-day simulation in 2 h. However, it was not possible to complete the 5- or 10-d simulation in 2 h using only 28 computing nodes (224 cores). Peak performance is obtained using about 20 CPU cores, which can predict dust storms in 6 h for 10 d ahead and in 3 h for 5 d in advance. This limitation is partially a result of cache and memory contention (43, 44). But it is more a reflection of spatial principles 2.b and 2.c that physical phenomena are temporally heterogeneous and the longer or bigger a dynamic process, the more exchanges occur among neighbors. Once again we find that increasing the temporal scope poses a major challenge to both the physical and computer sciences.

Connection Constraint. The execution time analysis is performed based on the result of one-day dust-storm simulations executed on two computing nodes with eight cores in each computing node. Different types of switches are used to connect computing nodes to enable them to communicate at various Internet speeds. To investigate the impact of a better connection performance, we calculate the network speedup ratio S as Eq. 3:

$$s = \Delta t_m / T_m, \quad [3]$$

where Δt_m is the decreased computing time of dust-storm simulation on m CPU cores when increasing the network connection speed from 100 Mbps to 1 Gbps, and T_m is the computing time when the network connection speed is 100 Mbps. For example, if 10 s is used for completing the simulation when using the 100 Mbps and 6 s when using 1 Gbps, then the Δt_m is 10 subtracted by 6 and s is 4 divided by 10. The result shows that the average performance improvement due to the network is more than 15.3%. The best performance improvement is 20% when 6 CPU cores are used (see Fig. 2F). Although it may not be important for long-term dust-storm forecasting, finer resolution predictions may take a longer time and can be very significant for real-time dust-storm simulations when results are desired in less than 1 h (42). It was observed that the network speedup ratio increases when the CPU core number increases with more CPU cores involved and more communication required, but speedup ratio decreases after 6 CPU cores. This is because two or more CPU cores of each computing node contributed to the dust simulation. It is part of the communication occurring among different cores within one computing node. Communication between these cores will not go through the Internet, therefore reducing the communication overhead (spatial principles 1.a.i and 1.a.ii). Thus, the network speedup ratio decreased after 6 CPU cores with 3 cores per computing node were involved as more communication took place within one computing node. And, spatial principle 1.a.i can be utilized to reduce the communication overhead because faster communication can be achieved by selecting closer computing nodes.

In this research, we identified and utilized spatial principles to design HPC arrangements and to parallelize simulation modes. The study was utilized in dust-storm research enabling scientists to predict dust storms at higher resolution (3 km \times 3 km) and for longer times (5–10 d) by adopting the proper number of CPU cores and decompositions of the domain to achieve best performance. Because of the massive need for physical science phenomena simulation and the prevalence of spatial principles in many phenomena, these parallelization and analyses methods can also be applied to the broader physical and social science domains, such as public health and economics.

Spatial Data Visualization

Physical science phenomena are inherently complex and multidimensional and therefore require visualization tools to help understand the physical processes and the underlying driving factors. Recently, data acquisition techniques and model simulations significantly increased the resolution and volume of spatial data. These two factors present great challenges for scientific visualization. To detect the movement patterns of dust storm for prediction and hazard mitigation, for example, researchers typically visualize simulated dust storms over a region with 4D dust density data for a defined period. With improvements in the climate models, spatial resolution can be refined to beyond 1/12 degrees (14) toward several kilometers. Similarly, the temporal resolution improves from daily to hourly for better prediction when spatial principles are applied. Consequently, visualization intensity increases by a factor of at least $4(\text{latitude}) \times 4(\text{longitude}) \times 24(\text{time}) = 384$ when compared to the original intensity for the same region. Such a dataset cannot be handled by a conventional visualization framework. Researchers need an advanced visualization tool to help solve these scientific problems by utilizing spatial principles.

To address these visualization challenges we designed our research program to identify where the bottlenecks occur and then provided an initial investigation on how spatial principles can be applied to address these problems. Dust-storm visualization is used as an example. Visualization of these events is critical to understanding the spatial distribution, detecting the temporal changes, and predicting the movements of dust particles that are used to inform the public about the development of dust storms. If the visualization of such dynamics is not straightforward, scientists may face a series of difficulties that are not exclusive to dust-storm prediction, but exist across the entire Earth system science domain. Factors such as the high complicity of dust density data derived from the climate model, the visualization modes used to represent the movement of dust particles, and the spatial heterogeneity during the interaction under the spatial principles govern the visualization capacity and performance.

To identify the bottlenecks within a visualization process, an experiment was conducted with a one core desktop machine using the dust density data produced by the models introduced in the previous section. The data used for 4D visualization are hourly dust particle density data generated by the NMM-dust. The original hourly data are stored in NetCDF files (a common scientific file format for model and observational data). The geographic coverage of the data is from 25.56°N to 41.48°N and from 123.00°W to 96.51°W. We focus on the data with this fixed spatial extent but varying spatial resolutions and time stamps. The visualization was implemented in the World Wind visualization environment. By increasing the volume of data used for volume rendering, our test found three extremes for a system to handle the data (Table 1). To solve these problems, researchers have developed four types of solutions including compression and simplification, multiresolution methods, memory external methods, and abstraction (45) by breaking large data into smaller regions to be handled by computing resources in parallel (46, 47). Also, parallel computing resources can concurrently handle the computing requirements to a limited extent. The reality is, however, that computing and

processing capacities have in many cases lagged behind increases in data volume. Besides parallel computing, visualization requires the consideration of spatial principles to solve spatial complexity issues. This situation is reflected in the coherence within visualization granularity among the understanding hierarchy, spatial data dissimilation and segmentation, and spatial-data-relationship preservation during the visualization process (48, 49).

Our strategy exploits spatial principle 1.a in the following aspects: (i) Data organization for facilitating fast access to large-scale spatially continuous data, for visualization through a multilevel octree structure (see 1.b). The data from the same nodes are geographically related (1.a.ii). (ii) Data distribution and aggregations. According to 1.a.iii, duplications are necessary for the final integration of distributed data. We handle the duplication needs by extending the boundary of each block and merging these boundaries in the later synchronization. This method is a balance between data redundancy and accuracy. (iii) Consideration of visual effects with respect to rendering algorithms. The volume rendering is best implemented when the spatial resolutions along each direction are the same. Thus an interpolation may be applied to regulate the spatial resolution and the block size with the underlying spatial continuities. And a levels of details (LOD) strategy was developed based on the octree structure to promote the hierarchical understanding (2.a) and reduce the complexity of visualizing large volume datasets.

We designed a framework (see Fig. 3A) with effective parallel computing and octree-based data organization to facilitate interactive volume rendering of multidimensional data. The first step in processing large volume datasets is to decompose the data to meet the physical memory configuration (44). Loading capacity is evaluated to obtain the maximum data block size. Decomposition divides the original data along latitude and longitude dimensions (see Fig. 3A). The pressure (vertical) dimension is not subdivided due to its much smaller size, which may influence the volume rendering process. The borders of decomposed data pieces have a buffer to avoid fuzzy visual results (1.a.iii). And an octree index is introduced to organize the data and control the rendering process. This framework has the following characteristics: (i) a simulated “out of core” strategy is used to load data dynamically, which makes the best use of physical memory and rendering capacity; (ii) a double multithreaded decomposition in the data distribution and visualization process, which balances the computing power with computing sources; and (iii) an adaptive query from an octree forms the LOD, which is designated to adapt the framework to different visualization platforms. Three tests of frames per second (FPS) are done to compare the improvement by introducing such a framework.

The first experiment assigns the rendering tasks to different numbers of threads to examine the role of multithreaded processing. The second experiment incorporates the octree index to accelerate the rendering speed. The dust density data are the same in the first two experiments, which have a vertex number of $32 \times 256 \times 128$. The third experiment found the maximum visualization capacity of this framework applied in these two systems, respectively.

The first experiment (Fig. 3B) found that multithreading alone cannot significantly improve real-time rendering. The second

experiment incorporated the octree index and found an increase in FPS for both the one-core and eight-core machine because the octree-based LOD derived by representation hierarchy (1.b) can provide an approximate visualization thus reducing the rendering intensity. In both experiments, the performance on the eight-core is better than that on the one-core machine. For the eight-core machine, the highest FPS appears when eight threads are present in both scenarios. For the one-core, the peak of FPS differs resulting in eight threads for octree based and 32 threads without octree, respectively. Because the neighboring cells need to communicate (1.a.iii), overdecomposition will reduce the rendering speed instead of improving performance. Therefore, eight threads could be a reasonable assignment that utilizes the rendering capacity.

The third experiment (Fig. 3C) found that the overall loading capacity exceeded the extreme value identified in Table 1. The FPS of the eight-core machine is reasonably good with respect to the increase of data volume. Confirming the results of the previous experiment, the communication among threads may influence the rendering performance, demonstrated by the fact that utilizing fewer threads can obtain the best performance, whereas more threads do not necessarily lead to an increase of FPS. Fig. 3C also shows that when data volume increases to about 1 GB, the visualization system of both machines fails again, due to the limited virtual memory.

This research advances the visualization of multidimensional data by providing a visualization tool for atmospheric scientists to vividly observe the dust model output and to more directly connect the parameters to the simulation results. Because of the nature of large data volume and spatial principles, the research can also be applied to other science domains, such as ecology and geology. The ability to explore the dynamics and spatial patterns behind physical processes can also be extended to other multidimensional domains.

More research is needed to leverage spatial principles to solve online visualization problems, concurrent massive user access, and display device issues adaptation so that physical science representations can be understood by the physical scientists more vividly (46, 50–53). However, a systematic solution is still needed, especially for the scalability and mobility for fast interactive spatial temporal visualizations according to the spatial principles 2.b and 2.c (53, 54).

Discussion and Conclusion

This paper defines spatial computing, the computing aspect of spatial CI, as the computing paradigm that leverages spatial principles to optimize distributed computing that enables physical science discoveries (55, 56). We conducted empirical studies to determine how spatial computing can facilitate the advancement of physical sciences. Using dust-storm research as an example, our research examples collectively demonstrate methods for identifying,

Table 1. Maximum capacity test for visualization

| Vertex number (Elevation× Latitude× Longitude) | Spatial resolution (Unit: degree) | Data volume (Unit: M) | Reason for system failure |
|---|---|-----------------------------|---|
| $32 \times 256 \times 128$ | ~0.12(lat), ~0.10(lon) | ~4 | Very slow interactive rendering speed |
| $64 \times 512 \times 256$ | ~0.06(lat), ~0.05(lon) | ~32 | Cannot manipulate data such as converting into pixel values |
| $128 \times 2,048 \times$ 1,024 | ~0.03(lat), ~0.025(lon) | ~1,024 | Cannot load data from the original file, I/O error |

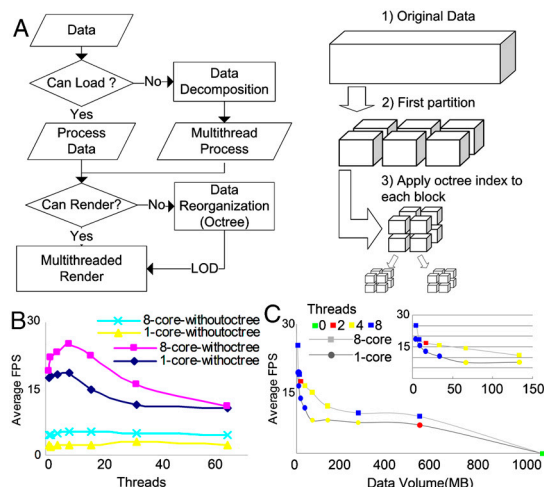


Fig. 3. Octree and multithreading techniques based on spatial principles are devised to enable visualizing large-volume spatiotemporal data.

analyzing, and utilizing spatial principles in spatial CI design for physical science applications. Specifically, (i) spatial principles can be utilized in computing to effectively locate, access, and utilize data resources, such as services, for deploying a spatial CI with good performance to solve physical science problems; (ii) spatial principles can be utilized in HPC to develop solutions for simulating physical science phenomena, providing more understanding of the past, and better prediction of the future; (iii) spatial computing can help leverage distributed computing power to enable visualization of data and research results for scientists, educators, and students to better understand multidimensional scientific phenomena.

It is not too bold to expect that spatial computing will provide an enabling technology for the new physical science frontier by contributing essential computing architectures, algorithms, and methodologies to construct the spatial CI (1) for solving problems with characteristics of data intensive, computing intensive,

spatiotemporal intensive, and concurrent intensive (4). It requires scientists, engineers, and educators from multiple domains to collaborate to solve fundamental problems (55), e.g., how to forecast high-resolution phenomena with broad geographic coverage for regional emergency responses, such as tsunamis. Although SDI starts as a good example of the construction of a spatial CI and we have seen notable successes from Google Earth, World Wind, and Bing Maps, the aforementioned challenges remain to be solved.

ACKNOWLEDGMENTS. Drs. Dawn Wright, Michael Goodchild, and anonymous reviewers provided insightful comments and guidance for us to complete the paper. Min Sun, Lizhi Miao, and Wenwen Li helped with paper preparation. Dr. Rob Raskin, Ms. Unche A. Saydahmat, and Mr. Steve McClure assisted with language polishing. We thank Drs. Ming Tsou and Shaowen Wang for helping conduct the first research example. This research is supported by NASA (NNX07AD99G) and the Federal Geographic Data Committee (G09AC00103).

- NRC (2009) *Informing Decisions in a Changing Climate* (National Academies Press, Washington, DC).
- Yang C, Wong D, Miao Q, Yang R (2010) *Advanced Geoinformation Science* (CRC Press, Boca Raton, FL).
- Wright DJ, Wang S (2011) The emergence of spatial cyberinfrastructure. *Proc Natl Acad Sci USA* 108:5488–5491.
- Yang C, Raskin R, Goodchild M, Gahegan M (2010) Geospatial cyberinfrastructure: Past, present, and future. *Comput Environ Urban* 34:264–277.
- Lautenbach CC (2006) The global earth observation system of systems: Science serving society. *Space Policy* 22:8–11.
- Gore A (1998) The digital earth: Understanding our planet in the 21st Century. speech given at the California Science Center, Los Angeles.
- Nebert DD, ed. (2004) *GSDI Cook Book Version 2.0* p 171.
- Ian Masser (2005) *GIS Worlds—Creating Spatial Data Infrastructures* (ERSI Press, Redlands, CA) p 312.
- Hornsby KS, Yuan M (2008) *Understanding Dynamics of Geographic Domains* (CRC Press, Boca Raton, FL) p 240.
- Sieber RE, Wellen CC, Jin Y (2011) Spatial cyberinfrastructures, ontologies, and the humanities. *Proc Natl Acad Sci USA* 108:5504–5509.
- Yang C, Wong D, Yang R, Kafatos M, Li Q (2005) Performance improving techniques in WebGIS. *Int J Geogr Inf Sci* 19:319–342.
- Yang C, Raskin R (2009) Introduction to distributed geographic information processing. *Int J Geogr Inf Sci* 23:553–560.
- Longley P, Goodchild MF, Maguire DJ, Rhind DW (2005) *Geographical Information Systems and Science* (Wiley, San Francisco, CA) p 497.
- Xie J, Yang C, Zhou B, Huang Q (2010) High performance computing for the simulation of dust storms. *Comput Environ Urban* 34:278–290.
- Yang C, Li W, Xie J, Zhou B (2008) Distributed geospatial information processing: Sharing earth science information to support Digital Earth. *Int J Digital Earth* 1:259–278.
- Cao Y (2007) Transportation routing with real-time events supported by grid computing. PhD dissertation (George Mason University, Fairfax, VA).
- Klemm R (2006) *Principles of Space-Time Adaptive Processing* (Institute of Engineering and Technology, London, UK).
- Huang Z, et al. (2011) Building the distributed geographic SQL workflow in the Grid environment. *Int J Geogr Inf Sci* 107.10.1080/13658816.2010.515947.
- Laube P, Duckham M, Croitoru A (2009) Distributed and mobile spatial computing. *Comput Environ Urban* 33:77–78.
- Hayes B (2008) Cloud computing. *Commun ACM* 51:9–11.
- Şen Z (2009) *Spatial Modeling Principles in Earth Sciences* (Springer, New York).
- De Smith MJ, Goodchild MF, Longley P (2007) *Geospatial Analysis: A Comprehensive Guide to Principles* (Troubador Publishing Ltd., Leicester, UK).
- Goodchild MF, Yuan M, Cova TJ (2007) Towards a general theory of geographic representation in GIS. *Int J Geogr Inf Sci* 21:239–260.
- Crawford D, et al. *Cyberinfrastructure Vision for 21st Century Discovery* (National Science Foundation, Washington, DC), National Science Foundation Publication CISE051203.
- Zaslavsky I, Memon A, Petropoulos M, Chaitan B (2003) Online querying of heterogeneous distributed spatial data on a grid. *Proceedings of Digital Earth* pp 813–823.
- Li Z, Yang C, Wu H, Li W, Miao L (2010) An optimized framework for seamlessly integrating OGC Web services to support geospatial sciences. *Int J Geogr Inf Sci* 10.1080/13658816.2010.484811.
- Yang P, Cao Y, Evans J (2007) WMS performance and client design principles. *J Geoinformation Sci Remote Sens* 44:320–333.
- Yang P, et al. (2007) The emerging concepts and applications of the spatial Web portal. *Photogramm Eng Rem S* 73:691–698.
- Fox G, et al. (1988) *Solving Problems on Concurrent Processors*, (Prentice Hall, Englewood Cliffs, NJ), Vol 1.
- Helly JJ, Kaufmann RS, Vernet M, Stephenson GR (2011) Spatial characterization of the meltwater field from icebergs in the Weddell Sea. *Proc Natl Acad Sci USA* 108:5492–5497.
- Zhang C, Zhao T, Li W (2010) Automatic search of geospatial features for disaster and emergency management. *Int J Appl Earth Obs* 12:409–418.
- Armstrong MP, Cowles M, Wang S (2005) Using a computational grid for geographic information analysis. *Prof Geogr* 57:365–375.
- Moscibroda T, Mutlu O (2007) Memory performance attacks: Denial of memory service in multi-core systems. *Proceedings of the 16th USENIX Security Symposium* (USENIX Association, Boston), pp 257–274.
- Janjic ZI, Gerrity JP, Jr, Nickovic S (2001) An alternative approach to nonhydrostatic modeling. *Mon Weather Rev* 129:1164–1178.
- Janjic ZI (2003) A nonhydrostatic model based on a new approach. *Meteorol Atmos Phys* 82:271–285.
- Janjic ZI, et al. (2005) High resolution applications of the WRF NMM. Extended Abstract. *21st Conference on Weather Analysis and Forecasting/17th Conference on Numerical Weather Prediction* (American Meteorological Society, Washington, DC).
- Purohit S, Kaginalkar A, Jindani I, Ratnam JV, Dash SK (1999) Development of parallel climate/forecast models on 100 GFlops PARAM computing systems. *Proceedings of the Eight ECMWF Workshop on the Use of Parallel Processors in Meteorology* (World Scientific, Reading, UK).
- Baillie C, Michalakes J, Skilinn R (1997) Regional weather modeling on parallel computers. *Parallel Comput* 23:2135–2142.
- Kozar C, Reilein R, Runger G (2001) Load imbalance aspects in atmosphere simulations. *Parallel Processing Workshops, 2001. International Conference on 3–7 Sept. 2001* (IEEE Computer Society Press, Valencia, Spain), pp 134–139.
- Natarajan C, Iyer R, Sharma S (1993) Experimental evaluation of performance and scalability of a multiprogrammed shared multiprocessor. *Proceedings of the 5th IEEE Symposium on Parallel and Distributed Processing* (IEEE Computer Society Press, Dallas, TX), pp 11–18.
- Nanjundiah RS (1998) Strategies for parallel implementation of a global spectral atmospheric general circulation model. *Proceedings of High Performance Computing, 1998. HIPC '98. 5th International Conference on 17–20, Dec. 1998* (IEEE Computer Society, Chennai, Madras, India), pp 452–458.
- Lenz CJ, et al. (2002) Meteo-GRID: World-wide local weather forecasts by GRID computing. *Proceedings of TERENA Networking Conference* (Elsevier Science, Limerick, Ireland).
- Smith AJ (1981) Internal scheduling and memory contention. *IEEE T Software Eng* 7:135–146.
- Denning PJ, Kahn KC (1976) An $L = S$ criterion for optimal multiprogramming. *Proceedings of International Symposium Computer Performance Modeling Measurement and Evaluation* (Harvard University, Cambridge, MA), pp 219–229.
- Joy I (2009) *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration, Mathematics and Visualization*, ed T Möller (Springer, Heidelberg), pp 285–302.
- Law CC, Schroeder WJ, Martin KM, Temkin J (1999) A multi-threaded streaming pipeline architecture for large structured data sets. *Proceedings of Visualization '99* (IEEE Computer Society Press, San Francisco, CA), pp 255–232.
- Brodie K, et al. (2004) Visualization in grid computing environments. *Proc of the Conference on Visualization '04* (IEEE Computer Society, Washington, DC), pp 155–162.
- Zhang C, Hammad A, Zayed TM, Wainer G, Pang H (2007) Cell-based representation and analysis of spatial resources in construction simulation. *Automat Constr* 16:436–448.
- Hildebrandt K, Polthier K, Preuss E (2005) Evolution of 3d curves under strict spatial constraints. *Proceedings of the Ninth International Conference on Computer Aided Design and Computer Graphics* (IEEE Computer Society, Washington, DC), pp 40–45.
- Ma KL, et al. (2007) Ultra-scale visualization: Research and education. *J Phys Conf Ser* 78:012088.
- Wu Q, Gao L, Chen Z, Zhu M (2009) Pipelining parallel image compositing and delivery for efficient remote visualization. *J Parallel Distr Com* 69:230–238.
- Huang J, Liu H, Beck M, Gao J, Moore T (2006) A distributed execution environment for large data visualization. *J Phys Conf Ser* 46:570–576.
- Sastry L, Fowler R, Nagella S, Churchill J (2009) Supporting distributed visualization services for high performance science and engineering applications—A service provider perspective. *Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid* (IEEE Computer Society, Washington, DC), pp 586–590.
- Greensky J, et al. (2008) Ubiquitous interactive visualization of 3D mantle convection using a Web-portal with Java and Ajax framework. *Visual Geosciences* 13:105–115.
- Poore BS (2011) Users as essential contributors to spatial cyberinfrastructures. *Proc Natl Acad Sci USA* 108:5510–5515.
- Wang S, Armstrong M (2009) A theoretical approach to the use of cyberinfrastructure in geographical analysis. *Int J Geogr Inf Sci* 23:169–193.