

Grid Computing and the TeraGrid GI Science Gateway

Kate Cowles

22S:295 High Performance Computing Seminar, Nov. 29, 2007

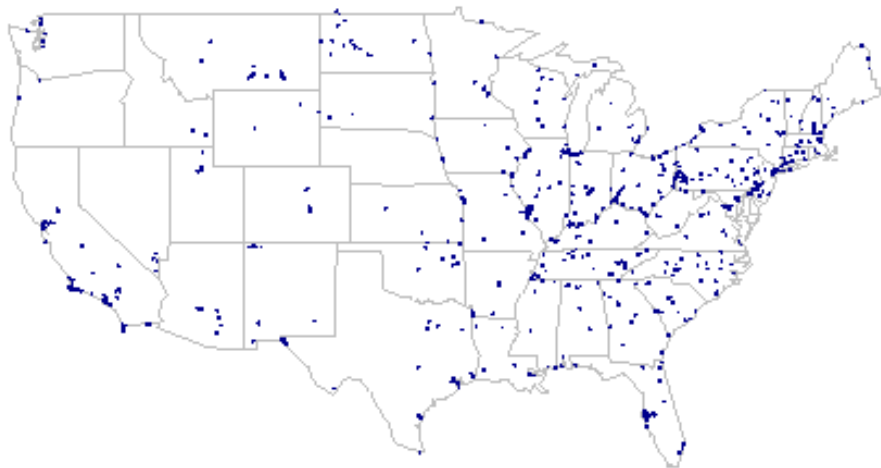
Outline

- 1 Example data
- 2 Grid computing
- 3 The TeraGrid
- 4 TeraGrid Science Gateways
- 5 GISolve

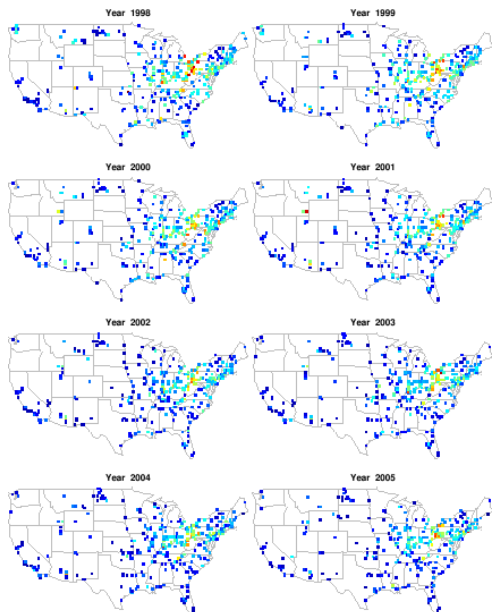
Outline

- 1 Example data
- 2 Grid computing
- 3 The TeraGrid
- 4 TeraGrid Science Gateways
- 5 GISolve

Sulfur dioxide concentration data from the EPA Air Quality System database



$n = 4711$ observations from 870 sites in the years 1998–2005



Outline

- 1 Example data
- 2 Grid computing**
- 3 The TeraGrid
- 4 TeraGrid Science Gateways
- 5 GISolve

Grid computing

- several different definitions, all involving distributed computing
- networking together computing clusters at different geographic locations to harness computing and storage resources

Outline

- 1 Example data
- 2 Grid computing
- 3 The TeraGrid**
- 4 TeraGrid Science Gateways
- 5 GISolve

The Teragrid

- the world's largest, most comprehensive distributed cyberinfrastructure for open scientific research
- began in 2001 when NSF awarded \$45 million to establish a Distributed Terascale Facility (DTF)
 - to NCSA, SDSC, Argonne National Laboratory, and the Center for Advanced Computing Research (CACR) at California Institute of Technology
- coordinated through the Grid Infrastructure Group (GIG) at the University of Chicago

TeraGrid, continued

- more than 250 teraflops of computing capability (May 2007)
 - teraflop: trillion (10^{12}) floating point operations per second
- more than 30 petabytes of online and archival data storage (May 2007)
 - petabyte: quadrillion (10^{15}) bytes
- rapid access and retrieval over high-performance networks.

TeraGrid Resource Provider sites

- Indiana University
- Oak Ridge National Laboratory
- National Center for Supercomputing Applications
- Pittsburgh Supercomputing Center
- Purdue University
- San Diego Supercomputer Center
- Texas Advanced Computing Center
- University of Chicago/Argonne National Laboratory
- the Joint Institute for Computational Sciences
- the Louisiana Optical Network Initiative
- the National Center for Atmospheric Research

Communication **among** TeraGrid sites

- Each resource provider maintains 10+ Gbps to one of three TeraGrid hubs (Chicago, Denver or Los Angeles).
- The hubs are interconnected via 10 Gbps lambdas (fiber-optic communications lines).



Outline

- 1 Example data
- 2 Grid computing
- 3 The TeraGrid
- 4 TeraGrid Science Gateways**
- 5 GISolve

TeraGrid Science Gateways

- enable users with a common scientific goal to use national resources through a common interface
- account management, accounting, certificates management, and user support is delegated to the gateway developers
- three common forms:
 - A gateway that is packaged as a web portal with users in front and TeraGrid services in back.
 - Grid-bridging Gateway: Science gateway is a mechanism to extend the reach of the community's existing Grid so it may use the resources of the TeraGrid.
 - A gateway that involves application programs running on users' machines (i.e. workstations and desktops) and accesses services in TeraGrid (and elsewhere).

Current TeraGrid Science Gateways

http://www.teragrid.org/programs/sci_gateways/

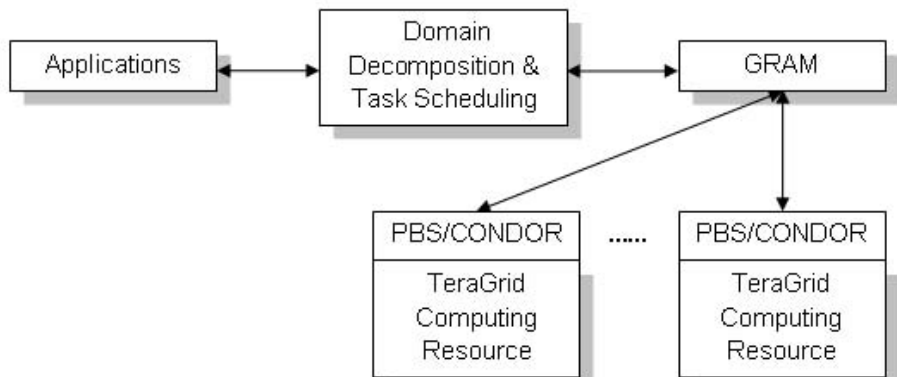
Outline

- 1 Example data
- 2 Grid computing
- 3 The TeraGrid
- 4 TeraGrid Science Gateways
- 5 GISolve**

- Geographic Information Science gateway
- web portal
- “The purpose of this project is to develop a TeraGrid Science Gateway toolkit for GIScience. Our gateway toolkit provides user-friendly capabilities for performing geographic information analysis using computational Grids, and help non-technical users directly benefit from accessing cyberinfrastructure capabilities.”
- current modules
 - 1 random spatial point generator
 - 2 distance-weighted interpolation of surfaces
 - 3 cluster detection algorithm (G_i^*)
 - 4 Bayesian geostatistical spatial model fitting using MCMC

Scheduling on TeraGrid Science Gateway web portals

- local job scheduler manages individual TeraGrid resource
 - Condor
 - Portable Batch System (PBS)
- Globus Resource Access and Management (GRAM)
 - interacts with local job schedulers to allocate computational resources for applications
 - monitors and controls computing processes
- user interactions with Science Gateways through TeraGrid software supporting Web Services Globus Toolkit



Geostatistical models

- natural and interpretable way to model spatial correlation for data measured at irregularly-spaced point sites
- correlation is a function of the distance, and possibly orientation, between sites

Simple geostatistical model with spatial correlation and additive measurement error

$$\mathbf{Y} \sim N\left(\mathbf{X}^T \boldsymbol{\beta}, \sigma_s^2 \Sigma(\phi) + \sigma_e^2 I\right)$$

- \mathbf{X} is a matrix of location-specific covariates
- $\boldsymbol{\beta}$ is a vector of coefficients to be estimated
- $\Sigma(\phi)$ is spatial correlation matrix
 - entries are calculated from correlation function
- σ_s^2 is spatial variance
- σ_e^2 is random variance (measurement error variance)
- I is identity matrix
- Bayesian model completed by specification of prior distributions on ϕ , σ_s^2 , σ_e^2 , and $\boldsymbol{\beta}$

Our alternative reparameterization

- facilitates prior specification and computing algorithm
- reparameterized covariance matrix

$$\sigma_s^2 \Sigma(\phi) + \sigma_e^2 I = \sigma_{tot}^2 [(1 - S) \Sigma(\phi) + S I]$$

where

$$\begin{aligned} \sigma_{tot}^2 &= \sigma_s^2 + \sigma_e^2 \\ S &= \frac{\sigma_e^2}{\sigma_s^2 + \sigma_e^2} \end{aligned}$$

Prior densities

- continuous uniform prior on ϕ
 - endpoints chosen to reflect belief as to largest and smallest possible distances at which spatial correlation could decay to 0
- joint prior on S and σ_{tot}^2 obtained by change-of-variable from inverse gamma priors on σ_e^2 and σ_s^2
- multivariate normal or flat prior on β

Spatiotemporal model with separable correlation structure

$$\mathbf{Y} \sim N\left(\mathbf{X}^T \boldsymbol{\beta}, \sigma_{tot}^2 \left\{ (1 - \mathbf{S}) \mathbf{K} [\boldsymbol{\Sigma}(\phi) \otimes \boldsymbol{\Sigma}(\rho)] \mathbf{K}^T + \mathbf{S} \mathbf{I} \right\}\right)$$

- where $\boldsymbol{\Sigma}(\rho)$ is an AR(1) matrix representing temporal correlation
- \mathbf{K} is a matrix of 1's and 0's that matches each observation Y_i with the correct row and column of $\boldsymbol{\Sigma}(\phi) \otimes \boldsymbol{\Sigma}(\rho)$
 - \mathbf{K} is not needed if data are "rectangular"
- prior on ρ uniform on $(-1, 1)$ or $(0, 1)$, slightly bounded away from endpoints

Computing algorithm in GISolve MCMC module

- very efficient MCMC computing algorithm that produces low autocorrelation in MCMC output
- computational bottleneck is linear algebra operations on big matrices, especially cholesky decomposition
- single-chain and multi-chain parallelization
 - linear algebra operations for each chain are parallelized using PlaPACK
 - all CPUs for an individual chain must be on same TeraGrid resource
 - multiple chains may be run simultaneously
 - “embarrassingly parallel”
 - different chains may be run on different TeraGrid resources
 - SPRNG used to make sure random number streams for different chains are independent

Challenges in TeraGrid implementation

- different TeraGrid sites have different software, libraries, and batch scheduling programs installed
- had to get PLAPACK installed and working on all sites where GISolve could be used

For more information

- details of model and algorithm currently implemented in GISolve are in Yan, Cowles, Wang, and Armstrong, *Statistics and Computing*, 2007
- extension of the sequential version of algorithm to handle prediction, areal data, fusion of areal and point source data, complicated spatial and nonspatial covariance structure
 - implemented in `ramps` package for R
 - explained in Cowles, Yan, and Smith 2007 and Smith, Yan, and Cowles 2007 (available as tech reports on stats dept web page)
 - not yet incorporated into GISolve

Using MCMC module in GISolve

- in advance of your session
 - get account on GISolve
 - request to reserve TeraGrid resources
- go to `www.gisolve.org` to log in
- upload file of spatiotemporal data you want to analyze
- upload configuration file
- select which TeraGrid partner sites you want to use
 - how many CPUs at each
 - how many parallel MCMC chains to run
 - number of iterations

- specify maximum wall clock time
 - must be long enough for the number of requested iterations to finish
 - must not run past the end of the reserved time on resource
- submit job
- click “Visualize output” to view plots of accumulating samples
- download zip files of plots and numeric output

Data file

- Data files must be plain text files
- First line is two integers:
 - number of rows of data
 - number of regression coefficients in model, including intercept
- data itself in rectangular format with the following columns (in order)
 - response variable
 - values of predictor variables, including a column of ones if intercept is required
 - x coordinate of spatial location (longitude)
 - y coordinate of spatial location (latitude)
 - integer representing measurement time

Data file example: sim2000.dat

```
2000 4
-0.0665 1 0.8056 0.9732 1 0.8056 0.9732 1
-3.0575 1 0.6244 0.4855 1 0.6244 0.4855 1
-2.1376 1 0.7375 0.3452 1 0.7375 0.3452 1
.
.
.
-9.9081 1 0.2165 0.4516 20 0.2165 0.4516 20
-9.9882 1 0.7621 0.0421 20 0.7621 0.0421 20
-8.8069 1 0.4893 0.4964 20 0.4893 0.4964 20
-8.6049 1 0.4142 0.9086 20 0.4142 0.9086 20
```

Configuration file

- specifies model to be fit
 - 1 choice among three spatial correlation functions
 - 2 specification of parameters of prior distributions on σ_{θ}^2 , σ_z^2 , ϕ , \mathbf{S} , ρ
- provides initial values for each MCMC chain

Configuration file, continued

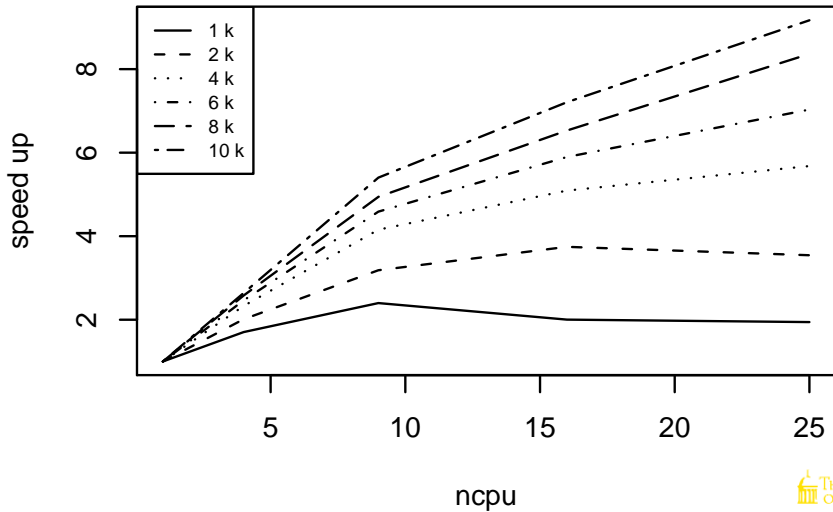
- content of individual lines
 - ① Correlation type (1 = spherical; 2 = exponential; 3 = Gaussian)
 - ② Distance metric (1 = great circle distance; 2 = euclidean distance)
 - ③ The unit of distance: for example, `distunit = 10` means that the distances are in 10s.
 - ④ $\alpha_e, \beta_e, \alpha_z, \beta_z$ (parameters of IG priors for σ_{2e} and σ_{2z})
 - ⑤ Left and right endpoints of uniform prior distribution for phi
 - ⑥ Left and right endpoints of support of distribution for S
 - ⑦ Left and right endpoints of uniform prior distribution for rho
 - ⑧ `block_size block_size_alg` (PLaPACK configuration; leave as in example)
 - ⑨ Chain index (from 0 to (number of chains - 1)) and initial values for phi, S, and rho
 - as many rows of this kind as there are chains

Configuration file example

```
1
2
1.0
0.5 0.5 0.5 0.5
0.05 2.0
0.01 0.99
0.01 0.99
200 400
0 0.5 0.75 0.75
1 1.0 0.5 0.5
2 1.5 0.25 0.25
```

Specifying number of CPUs and number of chains at each site

- number of CPUs is total number to be divided among all the chains at the site
- make number of CPUs *per chain* a perfect square to use PLAPACK efficiently
 - how big a perfect square determined by size of dataset (see graph of speedups in next slide)
- running parallel chains
 - helps in assessing convergence
 - generates more samples per unit time if CPUs are available
 - samples from different chains are independent
- efficient MCMC algorithm results in short burn-in, so a relatively small number of iterations are “wasted”



- choosing numbers of CPUs and chains
 - for dataset of 10000 observations, if you have 32 CPUs available at each of 3 sites, perhaps run 1 chain using 25 CPUs at each site
 - for dataset of 2000 observations, perhaps 3 chains at each site, each chain using 9 CPUs
- ability to extend chains from where they left off is being added