Using Open-Source Grid-Computing Technology to Improve Processing Time for Geospatial Syndromic Surveillance Data

Shaun Grannis MD MS,[1] Karen Olson PhD,[2] James Egg BS,[1] J. Marc Overhage MD PhD[1]

[1]The Regenstrief Institute, Inc. and Indiana University School of Medicine

[2]Children's Hospital Boston and Harvard Medical School

OBJECTIVE

We describe a method to perform computationally intensive analyses on large volumes of syndromic surveillance data using open-source grid computing technology.

BACKGROUND

Outbreak detection algorithms for syndromic surveillance data are becoming increasingly complex. Initial algorithms focused on temporal data but newer methods incorporate geospatial dimensions. As methods evolve, it is important to understand the effects on detection of both algorithm parameters and population characteristics. Intensive, iterative data analyses are required to accomplish this. Even with leading-edge computer hardware, it can take weeks or months[1] to complete analyses using advanced signal detection techniques such as the space-time scan statistic in the SaTScan™ program.[2]

Given the strategic significance and national security implications of timely and accurate detection, proper tools for studying and thus improving increasingly complex surveillance algorithms are warranted.

METHODS

The grid computing facility at Indiana University[3] (IU) was used to process large volumes of clinical data to study outbreak detection. IU's grid resources consist of 2 clusters, each containing 204 nodes. All computational nodes have two 2.4 GHz Pentium 4 Xeon CPU's with 2.5 GB RAM, and use high-speed data interconnects. The clusters run Red Hat Linux Advanced Server 3 operating system, PBS Pro 4.5 resource manager[4], and the open-source Maui job scheduler[5]. We use a Linux version of SaTScan 5.0 for outbreak detection.

Indianapolis emergency department surveillance data from 2001 to 2003 were geocoded and separated into one-week units by syndrome category. Each week was analyzed by syndrome for outbreaks by comparing it to the previous six weeks of control data using the SaTScan spatial detection algorithm. A custom batch job-submission program monitored the job queue on the cluster to maintain the maximum allowed number of pending jobs (70) in the queue. Aggregate datasets contained 4 to 150 weeks of encounter data. The overall processing time for each dataset and individual one-week unit runtimes were recorded.

RESULTS

Processing serially, as is typical in a PC environment, can be an inefficient way to handle large projects. The gap in runtime greatly increases as the dataset grows larger in size (Figure 1). Note that for the largest dataset, parallel processing time is less than that for some smaller datasets. This occurs when shared computing resources in the parallel environment free up, enabling more of the target jobs in the queue to be processed.

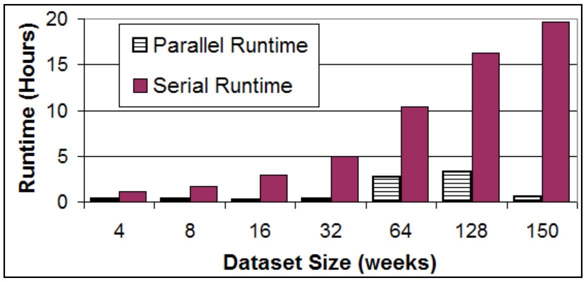

Figure 1: Processing time results

CONCLUSIONS

These results demonstrate that grid computing can dramatically reduce the time required to analyze syndromic surveillance data. This technology opens the door to new opportunities, including the potential to re-compute specific population models on-the-fly when an outbreak appears to be occurring. Further, many spatial scan statistics today are limited by massive search spaces. Grid computing's ability to "divide-and-conquer" these large tasks can provide the catalyst for future outbreak detection innovations. Further, the cost of grid computing is dropping dramatically. For example, a single 128-CPU desktop workstation is now available for $100,000. Consequently, what is considered leading edge technology today may soon be commonly available for syndromic surveillance tasks.

REFERENCES

[1] Olson KL, Grannis SJ, Mandl KD. Privacy protection versus cluster detection in spatial epidemiology. Manuscript submitted. 2005.

[2] Kulldorf, M. (2005). SaTScan 5 [Computer software]. Boston, MA: Harvard Medical School and Harvard Pilgrim Health Care. (www.satscan.org)

[3] Analysis and Visualization of Instrument-Driven Data (AVIDD) (http://support.uits.iu.edu/scripts/ose.cgi?almb.ose.help)

[4] Scapa J. (2004) PBS Pro (Version 4.5) [Computer software]. Troy, MI (www.altair.com/software/pbspro.htm)

[5] Maui Job Scheduler (2004) (www.clusterresources.com/products/maui/)