Automation or Interaction: What's best for big data?

Organizer:
***David Kenwright**, MRJ Technology Solutions, NASA Ames* Research Center

Panelists:
***David Banks**, Florida State University*
***Steve Bryson**, NASA Ames Research Center*
***Robert Haimes**, Massachusetts Institute of Technology*
***Robert van Liere**, CWI*
***Sam Uselton**, Lawrence Livermore National Laboratory*

INTRODUCTION

In the late 1800's telephone exchanges were manually operated and could only process a few callers a minute. As the volume of calls grew, a single operator could not handle the demand and manual exchanges gave way to automated ones. Today, operators still connect some calls, usually when the caller needs additional information (or money), but the vast majority can be handled by automated systems. History is littered with examples of systems that have become automated as technology improves.

This panel questions whether we, the visualization community, are on the right track by concentrating our research and development on interactive visualization tools and systems. After all, research programs like the Department of Energy's *Accelerated Strategic Computing Initiative (ASCI)* run computer simulations that produce terabytes of data every day. This raises the following questions:

- Is it feasible to analyze terabyte data sets using interactive techniques?
- Has visualization reached a level of maturity where most of the tasks can be automated?
- Will automatic feature detection tools be able to find all the interesting phenomena?

Our distinguished panelists will provide answers to these questions. They have been asked to "choose sides" to stimulate the discussion and to provoke controversy. Steve Bryson and Robert van Liere make strong cases for interactive visualization tools, while Robert Haimes and David Banks will tell us why automation is required for big data. Sam Uselton brings balance to the debate by suggesting that both automatic and interactive techniques will play important roles in understanding big data sets.

POSITION STATEMENTS

David Banks

"Automation Suffices for 80% of Visualization"

Interactive visualization would be essential to those scientists who pursue unfettered exploration of unfamiliar data, the scientists who discover new phenomena in their simulation that they never suspected were there, the scientists who like to try new tools that other people have created for their use. As many of us have experienced first-hand, these scientists exist in the realm of science fiction and PBS specials, not in real life.

There are two primary applications of computer graphics in scientific computing: debugging and presentation.

Tom Crockett (ICASE) champions the paradigm of visualization as a 3D print statement to let you quickly hunt down an offending segment of code. An interactive debugger is great for finding errors, but most people only use one as a last resort. The automatically-generated compiler messages catch the large fraction of simple bugs, and print statements reveal most of the others. In the same way, automatic visualization tools are well suited for debugging scientific codes. With datasets reaching the terabyte scale, a scientist could spend hours exploring isosurfaces, volume density mappings, or particle paths latent in a dataset. Interaction is the method of last resort.

Others consider visualization to be primarily a post-processing step to create a slick demo or a colorful poster or an animation for the Web. If the scientist's intent is to display certain features, then the visualization tool should be designed to locate them in the data. Research in visualization therefore includes characterizing discipline-specific features (tumors, blood vessels, vortices, shock surfaces, oil deposits) via robust

algorithms. Setting up the right viewpoint and lighting and layout is important in preparing images for public presentation, but this requires interaction for the art department rather than for the scientist.

Steve Bryson

"Show me that" vs. "What's there?"

It is a well-worn adage that the question you ask in large part determines the answer you will get. That is a fine thing if you are asking the right questions. But insight and discovery are driven as much by open-ended, curious exploration as by having your specific questions answered. This is not a place for the obvious ensuing philosophical discussion, but when your questions are very narrow exploration becomes much more difficult to perform. Automatic feature detection requires the framing of very specific questions: "show me the vortices" or "show me where a specific condition is satisfied". As an example, consider a room full of stuff. Automatic feature detection is like saying "show me the red boxes in the room". You'll find out where the red boxes are, but you'll miss knowledge of other objects in the room. Many more such specific questions are required for automatic feature detection techniques to give me a sense of all the objects in the room. On the other hand I could simply say "show me what's in the room".

Of course scientific visualization is not so simple, otherwise we would not have these annual conferences. Even physical simulation data can be very abstract, and there simply is no canonical or obvious way to "show me the data". Thus we have to ask questions of the data, in our business in the form of graphical represention of that data. Yes, the specifics of the graphical representation will determine the type of information we obtain on that data. But automatic feature detection asks very specific questions so that the results can be computed algorithmically and simply represented. Thus automated detection of features in a data set will always detect the features you ask for, and nothing else. If this is what you want then you are done.

But if you want to have a broad understanding of the data set, in particular if you want to understand **why** certain features are in the data set it is rarely sufficient to just display features. A sense of the data "around" the features is critical to understanding their context and often their cause. Knowing the vortices in a flow has use, but understanding why those vortices are where they are requires knowledge of the flow around them. Put another way, much scientific investigation is of phenomena that, while subject to local laws, are determined by global considerations. Fluid flow is a very common example: the existence of a vortex is due to the shape of the object that the flow is moving around. A somewhat global sense of the flow is required to understand the subsequent vortices.

Getting a global sense of data is difficult, particularly when the data is in three or more dimensional space and may have several interacting components. As is well known, scenes can quickly get very cluttered when many aspects of data are presented at the same time. Interactive techniques, where you have a graphical representation that you may move about in a data set in near real time, allows you to rapidly sample different regions of the data in different ways. To continue the flow example, observing interactive streamlines around a vortex can give great insight into the cause of the vortex. Interactivity is required to allow a sense of exploring the data. The more intuitive the interaction interface, the better the exploration will be. A rapid exploration capability allows you to get a general sense of the data, which provides a context for any features that may be detected in your exploration.

In some circumstances, you don't even know what specific questions to ask: science advances when new questions are thought up in response to new ways of seeing things. In these cases interactive exploration is a valuable tool to allow you to ask old questions in new ways. Streamlines of a vector field, which originally represented the paths of particles in a (steady) flow, can be used to study the behavior of, for example, the gradient of a scalar such as pressure. Exploration of a gradient field via streamlines can provide new insights such as the maxima and minima structure and so on. (OK, a weak example, but it's hard to think of non-trivial examples of fundamentally new questions!) While the same game can be played with feature detection techniques, e.g. by looking for the vortices in a gradient field, it is not quickly apparent what such features represent.

So we are presented with a spectrum: At one end automated feature detection techniques provide specific answers to narrow questions. If the question is exactly appropriate to your problem the automated feature detection may be all you need. At the other extreme, you may be exploring data in a simulation in which you have little understanding and don't know the interesting questions. In this case a suite of interactive visualization techniques will allow you to get a sense of the data and perhaps prompt interesting questions and understanding. In between, as in the example of flow around a complex object, feature detection techniques can give you a good starting point for detailed interactive exploration.

I'm reminded of the situation in robotics, where the initial hope was that robots could be completely

autonomous. This turned out to be somewhat beyond our reach in general situations, but high-level human control of robots has been very successful. This mix of automated and interactive activity is, I feel, very informative for our field. While one may argue that if we were just a little smarter we could automate everything, I feel that it is precisely at the frontiers of our understanding that scientific visualization has the greatest leverage, and it is here that we know the least about what questions to ask. This will always be the case.

Robert Haimes

Beyond stone knives and bearskins

Programs like the Accelerated Strategic Computing Initiative (ASCI) represent a tremendous growth of large-scale computing applied to the analysis of scientific problems. Most of the proposed ASCI simulations create output data sets containing billions of words of information (distributed on a 3-D mesh) for the results of a single steady-state run. Clearly, transient simulations of the same spatial fidelity stress any available computer resources. The sheer size of this data results in an exceedingly difficult and time consuming analysis process. The task of interrogation and interpretation of this information is required so that the knowledge contained within the simulation can be extracted.

Traditional interactive visualization probes the data in order to locate and identify physical phenomena. In order to find important flow features, users must interactively explore their data using one or more of the visualization tools (iso-surfaces, geometric cuts, streamline, and etc.). Scientists and engineers that use them on a regular basis have reported the following drawbacks:

- Exploration Time: Interactive exploration of large-scale 3-D data sets is laborious and consumes hours or days of the scientists/engineers time.
- Field Coverage: Interactive visualization techniques produce output based on local sample points in the grid or solution data. Important features may be missed if the user does not exhaustively search the data set.
- Non-specific: Interactive techniques usually reveal the behavior in the neighborhood of a feature rather than displaying the feature itself.
- Visual Clutter: After generating only a small number of visualization objects the display becomes cluttered and makes visual interpretation difficult.

It is clear that these tools do not directly answer the questions of the investigator. An expert is required to infer the underlying field topology from the imagery supplied. Getting a more specific answer is required. Direct, automated feature extraction has the following advantages over these exploratory visualization tools:

- Deterministic Algorithms: If there are no 'parameters' that the user need adjust, then no intervention is required.
- Fully Automated: The analysis can be done off-line (without a visualization subsystem). It can be used by other components in the analysis suite (i.e., directly by a solver to adapt the mesh to better resolve the feature).
- Local Analysis: These schemes, where possible, perform only local operations. Therefore, the computations for each cell are independent of any other cell and may be performed in parallel. This is clearly advantageous in distributed memory compute arenas.
- Data Reduction: The output geometry is several orders of magnitude smaller than the input data set. This is an important characteristic for the size of a resultant output. High fidelity spatial and temporal results of the feature extraction can be stored on disk. This is usually not possible for the entire transient simulation.
- Quantitative Information: Precise locations for the extracted features are provide. Also, classification and measures of strength can be reported.

A simple analogy can be drawn to any complex code. A large-scale program (that runs for more than a couple of seconds) may perform billions of integer and float-point calculations. It is not necessary to examine each operation to know that the program is running properly. There is usually some metric that the user of the program can use to determine the results. Even large, long running scientific simulations report integrated values to the user as some measure of goodness. Unfortunately, most of these measures are based on numerics and not physics. The physics can be examined by automatically extracting the features of interest and then answering the question: Is this what I expected?

Only when something happens outside our expectations (our analogous large-scale program produces unanticipated results) do we need to more closely examine the operations. Interactive visualization is only the debugger of our 3-D scientific simulation codes.

Robert van Liere

"Sorry, but I'm not really sure what I'm looking at."

The importance of data visualization is clearly recognized in scientific computing. Display of simulation results and interactive steering of computation require interactive exploration environments in which a user can see relationships and test hypotheses.

To support this claim I will discuss two examples. Both examples are motivated by the lack of knowledge of what is contained in the data. The first example is from flow visualization: the exploration of a very large turbulent data set from a direct numerical simulation. The second example is from cell biology: the exploration of cell components acquired from a confocal microscope.

The need for exploration environments will increase as models become more complex, simulation solutions become more detailed or acquisition devices become more powerful.

Sam Uselton

Best bets for big data

"Automation or Interaction, what's best for big data?" is the wrong question! The data doesn't know or care! Seriously, the question should be rephrased to focus on what is best for the USERS of big data. And that requires understanding what the users are trying to accomplish, and why large amounts of data are involved.

The first thing to notice is that there are many users with a wide variety of reasons for interest in large amounts of data. A single user's interest, even in a particular data set, may also vary drastically over time. I like to characterize one dimension of the variation in users purposes as ranging between "scientific" and "engineering" purposes. Engineering purposes are characterized by specific goals that result in precise answers to be extracted from the data. "Where in this lease should I drill to get the most oil?" "What angle of attack results in the largest lift to drag ratio for this aircraft?" Scientific purposes are characterized by vague, qualitative goals or extremely broad and general goals, which result in a desire to browse through the data looking for something unusual or different? "How does turbulence develop in originally laminar fluid flow?" "How did the universe evolve to produce galaxies and stars?" Remember that this is a continuum, not a binary classification. It is clear that automated answer finding is easier for questions at the engineering end of this spectrum than at the scientific end.

We are now producing and collecting data of many different kinds at a rate that precludes thorough interactive exploration as means for discovering the answers at the scientific end of the spectrum. Automatic methods suffer from typical computer "blind spots" - finding what they are directed to find, not everything the user might find interesting. Many people now favor using collections of tools that allow both kinds of activities. It is important to have such tools that "play well together." And that is not enough; we also need tools that are some new hybrid, using automatic methods to find less specific "things" in data sets, and suggesting places and dimensions in which interactive exploration is likely to be interesting.

BIOGRAPHIES

David Kenwright is a senior research scientist with MRJ Technology Solutions and works in the Data Analysis Group at NASA Ames Research Center. His current research interests include flow feature dectection, vector field topology, and biomimetics. He received his BE degree with first class honors in 1988 and his Ph.D. in mechanical engineering in 1994 from the University of Auckland, New Zealand.

Steve Bryson is a research scientist in the Numerical Aerodynamic Simulation Systems Division at NASA Ames Research Center and currently leads the Data Analysis group. He does research in the application of virtual reality techniques for scientific visualization, of which the virtual windtunnel is the main focus. He is the general co-chair of IEEE Visualization '99.

Robert Haimes is a principal research engineer in the Department of Aeronautics and Astronautics at the Massachusetts Institute of Technology. He is the author of a number of scientific visualization software toolkits in use worldwide, including Visual3 and pV3. His professional interests include computational fluid dynamics, turbomachinery, numerical algorithms, parallel and distributed programming, and scientific visualization.

Robert van Liere is head of a small interactive visualization and virtual reality research group at the Center for Mathematics and Computer Science, CWI, in Amsterdam. The group's research activities focus on computational steering, high-performance visualization, and virtual reality. Robert has been at the center for 12 years. Before that Robert worked at TNO, at Dutch applied research organization.

Sam Uselton is a computer scientist in the Center for Applied Scientific Computing (CASC), and leads the research efforts in data exploration. He received his B.A. in Mathematics and Economics in 1973 from the University of Texas at Austin. He earned his M.S. in 1976 and his Ph.D. in 1981, both from the University of Texas at Dallas. His current research interests include interactive methods of exploring very large scientific data sets, methods for evaluating visualizations and visualization systems, data fusion, comparative analysis methods, feature specification and detection, pattern recognition, innovative user interfaces, direct volume rendering, parallel rendering and realistic image synthesis.