A Distributed Co-operative Problem Solving Environment

Mark Walkley, Jason Wood, and Ken Brodlie

School of Computing, University of Leeds,
Leeds, LS2 9JT, UK.
{markw,jason,kwb}@comp.leeds.ac.uk
http://www.comp.leeds.ac.uk

Abstract. Scientific research is often multidisciplinary in nature and hence large projects are frequently collaborative with participants from several separate research centres. Rather than being restricted to infrequent dissemination of results and meetings a framework is described for embedding scientific computing applications within a collaborative problem solving environment. This allows users to combine their expertise in an interactive visual environment. Separate users can collaboratively steer and visualize data from a numerical simulation by embedding the simulation within the IRIS Explorer visualization system. By making use of this system the user has access to the COVISA toolkit which facilitates collaboration between separate users of IRIS Explorer. The flexibility of this system makes it straightforward to visualize and control as many aspects of the solution process as are desired.

1 Introduction

Major scientific research is today carried out by teams, often at geographically separate locations. A drawback to this typical scenario is that while results can be distributed and discussed, it is often difficult to interact constructively without physically meeting. This is particularly true of work involving numerical simulation where the collaborating researchers may include experts in the physics of the problem, the numerical model of the problem, and the visualization of the data produced by the numerical model. It is the aim of this work to provide a collaborative framework from within which the different researchers can combine their expertise and analyse the various aspects of the work.

Modern scientific computing applications typically produce far more data than can be easily handled without the use of further scientific tools. Effective visualization of this data allows the users to extract information of interest in an efficient manner, for example to quantify the performance of the algorithm or the quality of the solution produced. A number of visualization tools exists for processing of scientific data. These tools can broadly be split into two categories: turnkey systems, where the tool is generally tailored to a specific problem domain and offers a simple to use interface at the expense of being non extensible; and modular visualization environments (MVEs) which offer a more generic approach

to visualization, are more complex tools to use, but are user extensible. It is this second category of tools that is of most interest for this work since their general extensibility allows the addition of components, such as numerical simulation codes, into a single environment along with the provided visualization functionality. There are several commercial MVEs available, e.g. IRIS Explorer [1] and AVS [2], and several open source systems, e.g. OpenDX [3] and SCIRun [4]. All of these tools work along similar lines offering the user a library of modules, presented as graphical blocks, that represent a variety of functions. These modules can be placed onto a workspace and graphically connected to define the route of the data through the modules. The modules implement different algorithms changing the data from its raw state to some viewable image or geometry. In practice it is common to view this visualization as a post-processing stage in scientific computing; however this approach leads to a cycle of probably lengthy computational simulation time followed by relatively fast visualization of the data. In some cases it may have been apparent quite early on that modifications to the numerical algorithm were required but this is not possible once the simulation has begun.

Computational steering allows the user control of the simulation process from within the visualization environment and, simultaneously, access to the data produced by the simulation which can be visualized and analysed while the simulation is progressing. This allows parameters of the model to be changed during the simulation if necessary. Some previous work in this area has used a web-based interface [5] however, as previously discussed, the visual environment of IRIS Explorer, and systems such as SCIRun, naturally extend to this purpose.

Collaborative visualization allows geographically separate users to share data, and hence co-operate in its visualization and analysis. If the data directly from the simulation can be shared then users can apply different visualization techniques at each location rather than looking at a static display provided by one of the participants. This approach is obviously flexible and allows users with possibly different backgrounds and motivations to explore data in their own way. A number of projects have been undertaken to investigate collaborative visualization using a variety of systems. One of the simplest methods for collaboration is to replicate the user interface and then control access to the system through the use of a *token*. Only the user in possession of the token can make changes to the system, other collaborators must simply sit and watch until the token is released. This approach, while being simple to comprehend, limits the potential for users to visualize the data in their own way as well as taking part in the collaborative session. This mechanism is employed by the COVISE [6] visualization system to provide collaboration but in general can be implemented on top of any visualization tool through the use of application sharing technologies such as that contained within tools like NetMeeting [7]. An alternative approach is to add components to existing visualization systems that allow them to be used in a collaborative context. This is particularly applicable to the MVE class of systems due to their extensibility and open architecture. COVISA [8] was the first extension of a modular visualization environment, using IRIS Explorer, to

provide collaboration. It provides modules that allow the sharing of data and control between connected IRIS Explorer sessions, as well as providing a collaborative API for module writers. A similar approach using AVS Express was used in the MANICHORAL project [9].

In this work we will combine the advantages of visualization, computational steering and collaboration for scientific computing applications into a framework within IRIS Explorer that allows the user(s) simultaneous control over every aspect of the process. This approach provides detailed visual data on the performance of the algorithm and allows adjustments to be made to aspects of the algorithm while it is progressing. The combination of computational steering and collaborative visualization allows separate users access to both the data produced and the control of the simulation process, resulting in a collaborative problem solving environment within IRIS Explorer.

2   Numerical Application

Fluid transport, or convection, is an important physical process, for example in atmospheric dispersion [10], requiring expertise in both the physical processes involved and the numerical modeling of those processes. The numerical model combines transport of scalar chemical components with reaction between those components and is a challenging problem involving physical processes with widely different timescales. A prototype for such a model is that of passive scalar convection of a single component. Here this is modeled using a standard streamline-upwind finite element method [11]. The problem is typified by a convection velocity which directs the transport process, and the solution typically contains sharp layers dividing regions of different concentration.

The collaborating researchers will have different motivations in the analysis of data from such a model. For example, the chemist may wish to monitor the concentration of the component throughout the domain whilst the numerical analyst may wish to view the errors in the numerical solution at that point. In particular by monitoring the process as it proceeds and being able to interrogate the simulation in a variety of ways, it may become apparent quite early on that problems are occurring. Decisions can then be taken as to whether the simulation should be stopped and restarted or, more usefully, if parameters of the numerical model can be adjusted and the simulation continued. These issues may be resolved by taking a more flexible view of the visualization process and integrating the simulation within the visualization environment.

3   Visualization using IRIS Explorer

IRIS Explorer is a general-purpose modular visualization environment. A visualization *map* is built by combining *modules* into a pipeline through which the simulation data is passed and converted into a rendered image. A module *librarian* is available that contains many of the standard visualization techniques, such as isosurface computation or cutting planes. Used in the simplest way the data

set is supplied at the start of the dataflow pipeline and converted into a rendered image which is viewed at the other end. The disadvantage of this *standalone* approach is that visualization is a post-processing stage and that the overall work cycle is limited to successive computations followed by visualization.

IRIS Explorer is an open system and the user can also design their own visualization techniques and compile them into modules that can be included directly in the visual environment, and linked into the pipeline. This facility opens up the possibility of including modules that are not necessarily directly associated with visualization and here, in particular, modules containing numerical simulation algorithms.

4   Computational Steering

The aim of this work is to fully integrate the computation and visualization stages previously discussed and so allow synchronous computation and visualization. Here the numerical application is *wrapped* into an IRIS Explorer module and so can be included directly in the dataflow pipeline. An obvious advantage of such an approach is the immediate availability of the full visualization capabilities of IRIS Explorer for data generated during the computation. Further modules may be connected to the pipeline at any time allowing alternative, or more detailed, visualizations of the numerical model while the computation is taking place. A further advantage of including the numerical application as a module is that parameters can then be fed into the module from the IRIS Explorer environment which control its operation. The design of the simulation module allows the creation of a module interface and hence access to any parameters of interest. These parameters can be features of the numerical algorithm, e.g. error tolerance or timestep, or the physical model, e.g. density or convection velocity, as well as controlling the visualization process. This allows the users simultaneous and flexible control over the physical model, the numerical model and the data visualization.

The numerical application described in Section 2 is shown in Figure 1 together with the control panel interface used to steer the simulation and a visualization of a solution isosurface.

5   Collaborative Visualization and Steering

IRIS Explorer allows synchronous collaborative visualization through the COVISA modules [8]. These enable a team of people, at different internet locations, to link their individual IRIS Explorer dataflow pipelines. Essentially the data can flow between pipelines, as well as within each pipeline. Special 'Share' modules are wired by the researchers into their pipelines to achieve this data transfer. Different collaborative scenarios can then be programmed. For example, results generated in the pipeline of researcher A can be fed across to a (possibly different) pipeline of researcher B, as illustrated in Figure 2, for further analysis. Researchers with identical pipelines can share control of any module in the
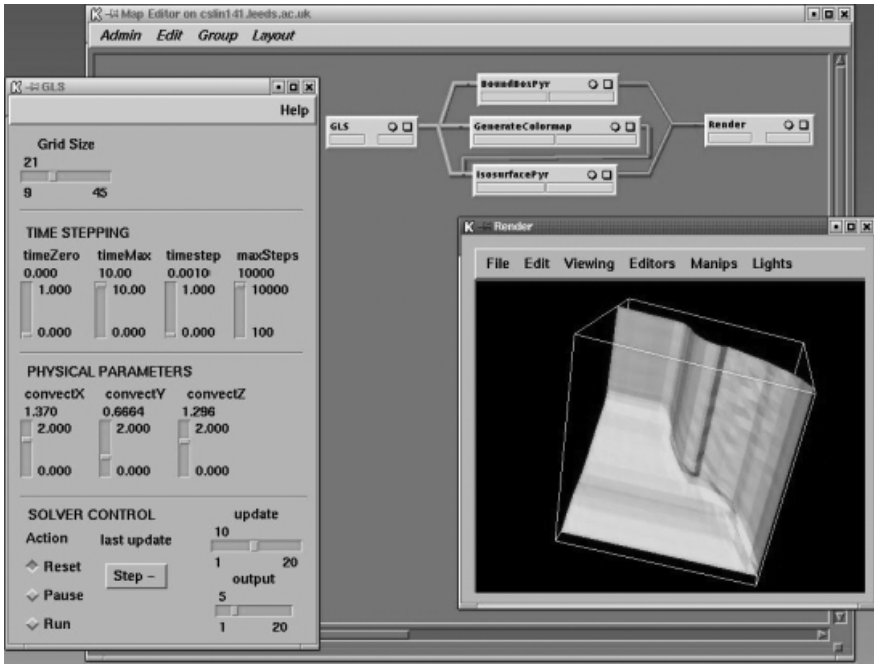
Fig. 1. IRIS Explorer map and visualization of the numerical application
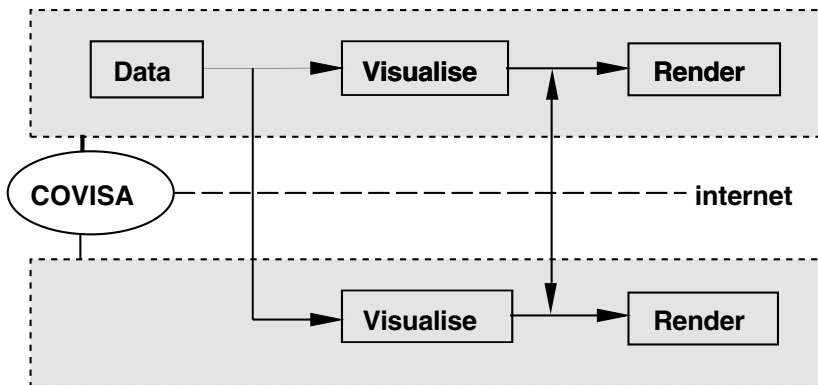

Fig. 2. Schematic of collaborative visualization

pipeline, by wiring in shared parameter modules at the appropriate point.

Here, since the numerical simulation functions as a module inside the visualization pipeline, the parameters used to steer the simulation can be supplied through the collaborative modules, as illustrated in Figure 3, allowing remote users to collaboratively steer the simulation as well as visualize the data produced by it. Several different types of data can be transferred across the network:
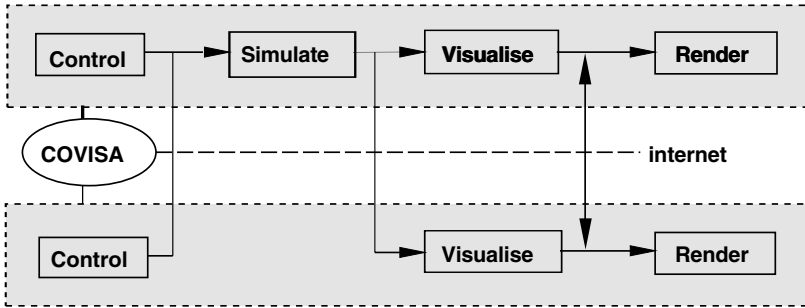

Fig. 3. Schematic of collaborative computational steering

single parameters, such as those used to steer the simulation; raw unstructured solution data from the numerical simulation; and rendered images, for example in the IRIS Explorer geometry format, or more standard formats such as gif files. Depending on the network bandwidth available, the most appropriate can be selected by the user to produce a responsive system. A useful collaborative scenario for the work considered here is that the numerical application, which is generally computationally intensive, can be run on the most powerful computer, while a *remote* user, who may have a relatively small computational resource, can control the simulation and receive data to visualize using the COVISA modules. In such a case it may be impractical for a user to monitor the process continuously. However it is also possible to use the collaborative modules to log in and out of a simulation, indeed the simulation can be run unmanned and the collaborative modules used to connect, or reconnect, a remote user. The user can monitor progress periodically and steer the simulation when necessary. This also avoids data having to be sent across the network unless it is required by the user. Note that computationally intensive visualization operations, such as isosurface computation, can also be performed on the most suitable platform.

Figure 4 shows a collaborative visualization session between two users for the application described in Section 2. The first, shown in Figure 4(a), is hosting the application and viewing an estimate of the error in the solution. Tetrahedral cells with an error exceeding a specified tolerance are visualized. The data from the simulation is shared with a second user, shown in Figure 4(b), who is viewing an isosurface of the computed solution. The parameters that control the simulation are also shared collaboratively allowing either user to steer the computation.

(a) User A running the simulation and visualizing the error

(b) User B collaboratively controlling the simulation and visualizing the solution

Fig. 4. A collaborative session using IRIS Explorer

6   Conclusions

It has been shown how IRIS Explorer can be used as a problem solving environment for numerical simulations. By using the collaborative capabilities of the COVISA modules this is extended to allow steering and visualization of the simulation by colleagues at geographically separate locations. This interactive environment allows researchers to share not only results but also experience of the various aspects of the numerical model in a simple and effective manner. The collaborative scenario developed can be tailored to the resources available, both in terms of network bandwidth and computing power, by choosing which parts of the computation are performed on each machine and the level of data that is shared collaboratively.

Modern scientific computing applications are typically highly computationally intensive and it is unrealistic to assume that they can always be run in a interactive manner. However the system developed will naturally extend to the unmanned mode described in Section 5. The collaborative connection can be used by a remote user to log in and out of the simulation and once connected the user can visualize the data produced in any manner desired and steer the simulation as before.

The computational Grid represents a new paradigm for distributed computing and recent research in computational steering and simulation is frequently concerned with allowing all, or parts, of the computation to be distributed in this way [5, 12]. Our current work in this area is focusing on extending the application to use the Globus *middleware* [13] to allow the computing to be spread over a network of computational resources. The scenario under development, depicted in Fig. 5, will use Globus to distribute the computationally intensive numerical
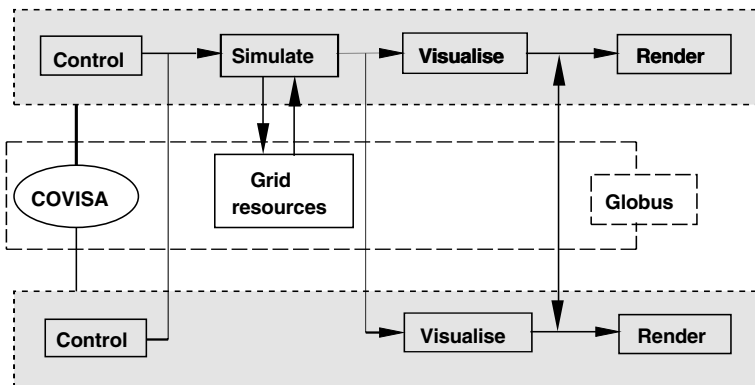

Fig. 5. Schematic of grid-based collaborative computational steering

simulation, with an interface from IRIS Explorer allowing collaborative steering and visualization of the process to be performed.

References

1. Foulser, D.: IRIS Explorer: A Framework for Investigation. Computer Graphics. 29(2) (1995) 13-16
2. Advanced Visual Systems: http://www.avs.com/
3. Open Visualization Data Explorer: http://www.opendx.org/
4. Miller, M., Hansen, C., Johnson, C.: Simulation Steering with SCIRun in a Distributed Environment. Lecture Notes in Computer Science. Springer-Verlag (1998)
5. Allen, G., Benger, W., Goodale, T., Hege, H.-C., Lanfermann, G., Merzky, A., Radke, T., Seidel, E., Shalf, J.: The Cactus Code: A Problem Solving Environment for the Grid. In: Proceedings of the Ninth IEEE International Symposium on High Performance Distributed Computing (2000)
6. COVISE: http://www.hlrs.de/organization/vis/covise/
7. NetMeeting: http://www.microsoft.com/windows/netmeeting/default.asp
8. Wood, J., Wright, H., Brodlie, K.W.: Collaborative Visualization. In: IEEE Visualization Conference (1997) 253-260
9. Duce, D.A., Gallop, J.R., Johnson, I.J., Robinson, K., Seelig, C.D., Cooper, C.S.: Distributed Cooperative Visualization - The MANICHORAL Approach. In: Eurographics-UK. Leeds (1998)
10. Hart, G., Tomlin, A., Smith, J., Berzins, M.: Multi-scale Atmospheric Dispersion Modelling by the Use of Adaptive Gridding Techniques. Environmental Monitoring and Assessment. 52 (1998) 225-228
11. Johnson, C.: The Finite Element Method. Wiley (1990)
12. Miller, M., Moulding, C., Dongarra, J., Johnson C.R.: Grid-enabling Problem Solving Environments: A Case Study of SCIRun and Netsolv. In: Proceedings of HPC 2001 (2001)
13. The Globus Project: http://www.globus.org