Programming Experience

Malcolm Roberts

February 6, 2015

1 Programming Experience

My research involves the application of high-performance computing to understanding turbulence. Turbulent systems exhibit a very large number of degrees of freedom and time-scales, requiring large amounts of both RAM and processing power, which has driven the development of both exa-scale computing and numerical methods. Since direct numerical simulations of turbulence require computing power that will remain unavailable for at least 100 years (assuming Moore's law holds), subgrid models, such as the multispectral method which has been the focus of my research in graduate school, allow one to determine approximate solutions using present-day hardware.

Most of my programming has been done in `C++`, producing serial code to numerically solve dependent dynamical systems. These were run mostly in grid computing environments, such as the University of Alberta numerical and statistical servers, the MARC Linux Cluster in Germany, and the western Canadian WestGrid system. During my work I have also developed a good working knowledge of scripting and version control systems (CVS, Subversion, and Mercurial). I am also familiar with TEX/LATEX and Asymptote, a vector graphics language designed for scientific figures.

For the problems on which I have worked, the most important part of programming has been the development of a mathematical description of the problem; by understanding which aspects of the system are relevant and which are not, one can reduce the difficulty of the problem and save time for both the programmer and the computer itself. My understanding of the mathematics behind the algorithms allows me to design a more efficient program which can, in some cases, reduce computational complexity by orders of magnitude.

2 Projects

- Implicitly dealiased convolution routines for grid computing. `C++`, `OpenMPI`
  2012-01 to present

  This project extends the method of implicit dealiasing [Bowman & Roberts 2011] to distributed memory architectures. We hope to use lower memory requirements of the implicit method to decrease the communication cost as compared to conventional explicit zero-padding of phase-shift dealiasing techniques.

- Multi-threaded version of implicit dealiased routines. `C++`, `OpenMP`,2011-11 to 2012-03

  This project called for the modification of the implicitly dealiased convolution routines in `fftw++` using FFTW's built-in parallel routines and `OpenMP`. I was able to reach between 80% and 90% efficiency (i.e. a speed up of a factor between 1.6 and 1.8 with two threads, or between 3.2 and 3.8 with four threads). This project is open-source.

- `VORT`, `C++`, 2003-10 to 2005-08

  I created this project to analyse the output of a simulation of two-dimensional turbulence and detects vortices, which are coherent circular structures in the flow field. This output is used to determine the rate of decay (due to collisions and viscous effects) of vortices as a step towards an "atomic" description of turbulence.

- `TRIAD`, `C++` 1998-08 to present

  `TRIAD` is a high-performance numerical integrator built for serial simulations of time-dependent problems. Numerical integrators allow one to take a continuous problem (such as determining the motion of the planets) and produce an approximate, discrete solution without having to solve the system analytically. My involvement in this project was to generalise the Runge-Kutta integrator class to allow easier implementation of new methods, improve the dynamic time-stepping routine, and to add the `MultiIntegrator/MultiProblem` which was used in the `GOYSR` and `MDNS` projects listed below. `TRIAD` is used as the back-end for `GOY`, `GOYSR`, `DNS`, `SRDNS`, and `MDNS` projects listed below.

  We hope to release `TRIAD` under the LGPL later this year.

- `GOY`, `C++` 2006-11 to 2009-08

  Shell models of the Gledzer, Ohkitana, and Yamada (GOY) type are one-dimensional toy models of turbulence which allow us to test ideas on simple systems before moving on to more complicated problems. `GOY` allows us to simulate such systems, which, while not exhibiting a great many degrees of freedom, are nevertheless difficult to simulate due to the great disparity of time-scales.

- `GOYSR`, `C++` 2005-06 to 2011-04

  This project uses the method of multispectral reduction (as developed in my masters thesis) to extend `GOY` to a hierarchy of grids, and makes use of the `MultiIntegrator/MultiProblem` class in `TRIAD`. The goal of this project was to simulate shell models of turbulence using a variable resolution while maintaining the important statistical-mechanical properties of the original system. `GOYSR` was a test-bed for ideas which formed the basis of `MDNS`, described below. This project formed the core of my masters dissertation [Roberts 2006].

- `FFTW++`, `C++` 2010-05 to present

  Convolutions are binary operations on arrays or functions which are important in such fields as statistical correlation analysis, filtering, and simulations of nonlinear systems. A naïve implementation of this algorithm takes $\mathcal{O}(n^2)$ operations, but this can be reduced to $\mathcal{O}(n \log n)$ by making use of the convolution theorem and fast Fourier transforms (FFTs). However, one must zero-pad these transforms in order to remove aliasing errors. In `FFTW++`, I implemented an implicit-padding algorithm that greatly reduces the memory requirements as compared with conventional techniques, while simultaneously reducing the computational time for multi-dimensional convolutions. These routines are described in detail in [Bowman & Roberts 2011] and [Roberts & Bowman 2011].

  `FFTW++` is available under the LGPL at `fftwpp.sourceforge.net`.

- `DNS`, `C++` 2009-08 to present

  The Navier–Stokes equations offer a good description of the motion of fluids, and can be numerically solved via the pseudospectral method. The most computationally expensive part of this method is the calculation of a convolutive term; following the development of `FFTW++`, we developed `DNS` to make advantages of this new technique for simulations of two-dimensional turbulence. This project uses `TRIAD` as a back-end for time-stepping, and forms the basis for the `SRDNS` and `MDNS` projects described below.

- `SRDNS`, `C++` 2009-03 to present

  This project is the first implementation of the method of spectral reduction which makes use of FFTs, allowing one to perform high-resolution simulations of two-dimensional turbulence using fewer data points than a straight-forward pseudospectral method such as `DNS`. The disadvantage of this technique is that large-wavelength waves are decimated to the same degree as short-wavelength waves, which results in increased error at the large scales. Full details are given in [Bowman & Roberts 2012]. This project uses special convolution routines based on `FFTW++` to calculate the nonlinear term in the evolution equations.

- `MDNS`, `C++` 2009-10 to present

  This project is the result of the application of the multispectral method, first developed for shell models in `GOYSR`, to two-dimensional Navier–Stokes turbulence. The application of the multispectral method to two-dimensional turbulence is complicated by the presence of multiple conserved invariants and inconvenient grid geometry as compared with shell models of turbulence. This project forms the core of my doctoral work [Roberts 2011].

References

[Bowman & Roberts 2011] J. C. Bowman & M. Roberts, SIAM J. Sci. Comput., **33**:386, 2011.

[Bowman & Roberts 2012] J. C. Bowman & M. Roberts, Communications in Nonlinear Science and Numerical Simulation, **17**:2008, 2012.

[Roberts & Bowman 2011] M. Roberts & J. C. Bowman, Journal of Physics: Conference Series, **318**:072037, 2011.

[Roberts 2006] M. I. W. Roberts, "A multi-spectral decimation scheme for turbulence simulations," Master's thesis, University of Alberta, 2006, `http://www.math.ualberta.ca/~bowman/group/roberts_msc.pdf`.

[Roberts 2011] M. Roberts, *Multispectral Reduction of Two-Dimensional Turbulence*, Ph.D. thesis, University of Alberta, Edmonton, AB, Canada, 2011, `http://www.math.ualberta.ca/~bowman/group/roberts_phd.pdf`.