Interpolating Wavelet Optimised Finite Differences

Jacques du Toit,* Numerical Algorithms Group (NAG) Ltd.

1 Background

A common technique for the efficient solution of PDEs is grid adaptivity. The idea is to devote more grid points to those regions of the domain where the solution needs them. Typically these would be regions where the solution has steep gradients, strong non-linearly or has other interesting features.

While the idea is simple, the practicalities are more involved. How does one decide on "steepness?" How should one use this information to place new points?

Dempster and Carton de Wiart [1] recently introduced a novel approach to this problem. They use ideas from wavelet compression theory, which gives rise to the name Interpolating Wavelet Optimised Finite Difference (IWOFD). A well-known wavelet (Donoho's wavelet) is taken and used to approximate the solution using *wavelet thresholding*. If the approximation is good, then it indicates that points can be removed from the grid. If the approximation is poor, then grid points should be inserted. The locations of the points to be added or removed is given by the wavelet. A finite-difference operator is then defined on this non-uniform grid, and the PDE is solved using standard finite difference methods. Note that IWOFD addresses grid adaptivity only – it does not cover time stepping.

NAG has received access to the IWOFD code, consisting of around 5000 lines of C/C++. Initial benchmarking suggests that the code is not implemented in an optimal way, with the biggest bottleneck appearing to be memory accesses, possibly stemming from inefficient data structures that are used to hold the grid and its various refinements.

2 Proposal

There are two main aspects to the proposed MSc project: implementation and research.

2.1 Implementation

The aim is to produce a more performant implementation of the IWOFD algorithm.

- Study the IWOFD method and the code
- Design an appropriate set of benchmark problems on which to test the method (3 to 5 problems should be sufficient). These can be the problems in [1] or other problems (not necessarily from finance), but thought should be given to finding problems that are likely to stress the method (i.e. strongly non-linear time dependent local features). Free boundary problems should be included in this set, as well as problems that require implicit time steppers.
- Compute reference solutions to the benchmark problems using whatever software package, discretisation method and time stepper is most appropriate.

*jacques@nag.co.uk

- Analyse the performance of the current version of IWOFD on the benchmark problems
- Create a more performant software implementation of IWOFD. There is a strong preference for this implementation to be done in C++ or Fortran.

2.2 Research

The aim here is to evaluate and hopfully improve the IWOFD algorithm, and to explore its applicability in other settings.

- Propose possible improvements to the IWOFD algorithm and test these ideas on the benchmark problems
- Investigate how to apply Geometric Multigrid to IWOFD to speed up implicit timesteppers. Existing GMG packages and codes (for example from mgnet.org) can be tested, but the student is also encouraged to come up with their own ideas.
- Investigate whether the ideas of IWOFD can be translated to a finite element setting in order to produce a spatially adaptive FE method. If the ideas do translate, then an initial implementation should be made and tested on some problems. This need not be a performant implementation and can be in the langauage/environment of the student's choice.

3 NAG Involvement

NAG will host the student for the majority of the project. The student will spend time in the Manchester and Oxford offices. The majority of time will most likely be spent in Oxford. The student will receive support from several members of staff with expertise in ODEs, PDEs, FE methods and adaptive meshing. NAG will cover the student's travel expenses and accommodation, and will provide a living allowance.

Depending on the results and progress made, NAG will also arrange input from relevant customers who are most likely to be in the investment banking industry.

4 Student Requirements

The main output of the project is a performant implementation of the current IWOFD algorithm. While it does not need to be optimised to the n-th degree, the implementation should be cache and memory efficient and should exhibit decent behaviour when probed with a performance profiler. There are many people at NAG who can (and will) help the student with this, but these are somewhat advanced topics and it is important that the student has strong programming skills and is comfortable programming in C++ or Fortran. The student should also be happy to spend a significant portion of the project in the UK.

Note that this is a provisional project proposal which may change (major changes are not anticipated).

References

[1] Carton de Wiart, B, and Dempster, M. A. H. (2011) Wavelet optimized valuation of financial derivatives. *Intl. J. Th. App. Finance* Vol 14, No 7 (1113–1137).