Undergraduate projects

Simon Shaw
people.brunel.ac.uk/~icsrsss

July 22, 2015

Contents

1 Print quality source

Don't print the web page – the quality of the math formatting is not good. Instead print this PDF file: `people.brunel.ac.uk/~icsrsss/modules/fyp/topics/home.pdf`.

2 Current projects

Currently available projects are listed in the following sections. Each one indicates whether or not it is purely theory, or a mix of theory and computing. In some cases the mix between computer programming and theory can be varied to suit the student's preference and/or course but the computing aspect of the project should not be regarded as optional. For these projects matlab is probably the best choice but if you have some Java, C/C++ or other knowledge then you may well be able to use those also. This will need to be discussed.

Note that a significant number of marks are allocated or closely connected to the quality of the written document. For a guide on how to 'write mathematics' at a professional level see

Higham's book: [7].

3 Finance and business topics:

3.1 Four Projects on the Numerical Solution of the Black-Scholes Partial Differential Equation

In the 'Black-Scholes world' the value of an option can be found by solving the *Black-Scholes partial differential equation* (BSPDE). Although exact solutions are known for the simplest types of option (European calls and puts for example), the BSPDE cannot be solved analytically for many types of *exotic option*. In such cases it needs to be solved numerically using, say, finite difference methods and matlab. The main reference for this project is the book [9] and, if you are taking it, the module MA3667 (the MA3976 replacement).

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Graphical illustration of prices for vanilla call and put options as given, for example, at `en.wikipedia.org/wiki/Black-scholes`.
- Finite difference approximation of the BSPDE. Matlab implementation and verification of accuracy using options for which the exact solutions are known.
- High quality graphical output from the BSPDE solver and its extension to various option strategies (e.g. straddles, strangles and butterfly spreads).
- Successful description and implementation of the numerical solution for one of the following.
  - An American put option, e.g. `en.wikipedia.org/wiki/American_option`
  - A compound option, e.g. `en.wikipedia.org/wiki/Compound_option`
  - A chooser option, e.g. `en.wikipedia.org/wiki/Chooser_option`
  - Transaction cost inclusion via the Hoggard-Whalley-Wilmott equation, e.g. `www.springerlink.com/content/b4236237786qx661/?MUD=MP` (but you may need to be logged in at Brunel to download it).

  The book [9] contains details of each of these. In addition, Geske's original paper on compound option valuation is here: `bbs.cenet.org.cn/uploadImages/20035291315398755.pdf`, and some other notes are here, `www.math.ust.hk/~maykwok/piblications/Articles/comp\%20option.pdf`, and here, `www.bus.lsu.edu/academics/finance/faculty/dchance/Instructional/TN98-05.pdf`. However, bear in mind that the main thrust behind these projects are numerical implementations of solution algorithms in matlab.
- For very high marks you will cover advanced material such as, for example, detailed analyses of the numerical methods, dealing with time varying interest rates and volatility, quantitative assessment of the effect of boundary truncation and so on. This type of advanced study will be agreed on a case-by-case basis between the student and the supervisor.

Involves: theory and computing.

3.2 Binomial Tree Pricing of Financial Options

Binomial trees provide an approximate means of pricing financial options that does not involve solving the Black-Scholes partial differential equation. The basic idea is to build a recombining binary tree of possible asset prices at discrete times that extend from now until the expiry date of the option. At expiry the value of the option (the known *payoff*) is calculated at the tree's final node layer, and then *discounted risk-neutral expectation* is used to recursively value the option backward in time until the current time is reached. The main references for this project are the books [1], [9], the paper [5] and, if you are taking it, the module MA3667 (the MA3976 replacement). You can also look at `en.wikipedia.org/wiki/Binomial_options_pricing_model` to get a flavour of what is involved.

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Graphical illustration of prices for vanilla call and put options.
- A recap of the Black-Scholes theory, the partial differential equation and its solution as a risk-neutral discounted expectation.
- A description of the binomial world and the population of the tree with log-normal asset prices. Implementation (e.g. matlab) is required.
- An implementation of backward recursion to determine the current value of a European option given the payoff function. Your code should generate high quality graphical output.
- For very high marks you will decribe and implement the extension of this theory to American options and also use your computer progam to estimate all of the so-called 'greeks'.

Involves: theory and computing.

3.3 Implied and Historical Volatility Estimation

In the Black Scholes theory of option pricing the price of the underlying asset is assumed to follow a lognormal random walk with the 'randomness' controlled by a parameter, σ, called the volatility. The value of this parameter is important for the accuracy of the option price but is not observable in the market. In this project you will study two common ways to obtain this parameter for a given asset. The first is the use of Newton's method where, given the option price and all other observable data, the volatility is given by finding the root of a nonlinear equation (see e.g. `en.wikipedia.org/wiki/Implied_volatility`). The second is the use of historical asset price data where one hopes to extract this volatility as a standard deviation of past prices. The main starting point is the book [6] and we note that historical data adequate for the purposes of this project can be obtained from, for example, `uk.finance.yahoo.com/q/hp?s=VOD`

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Comprehension of the asset price process used in the Black-Scholes theory and of its lognormal statistics.
- Cogent discussion of Newton's method for implied volatility and a demonstrably correct matlab implementation.

- Acquisition of a selection of historical data and a matlab implementations of historical volatility calculations.
- For very high marks you will give the theoretical details behind the robustness and convergence of Newton's mathod for this problem as well as provide confidence intervals and further theoretical developments for the historical voilatility calculation.

Involves: theory and computing.

4 Generic applied mathematics:

4.1 Theory and animation of planetary systems

The orbit of the earth around the sun is described by the branch of mechanics known as *Central Force Theory* and is encapsulated in Kepler's laws. Analytical solutions are known for the case of one large body (the sun) and one small body (the earth). When the other planets are brought into the picture the differential equations of Newton's theory of gravity can, in general, no longer be solved analytically and numerical methods (using, for example, matlab) must be used. The simulated solutions can then be used to produce animations. See the links at the bottom of `www.ams.org/featurecolumn/archive/orbits1.html` for some examples of what you can achieve.

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Coverage of classical central force theory and Kepler's laws (e.g. sun and earth), together with computer animations.
- Extension of the theory to 'many (N) body problems' in two-dimensions, each body having a different mass and position.
- Numerical simulation of the N-body problem with computer animations.
- For very high marks you will extend the N-body theory, algorithm and graphics to 3D, use an implicit time stepper and compare its performance to an explicit one, and you will illustrate the effect of discretisation stability and error on the physics.

Involves: theory and computing.

4.2 Theory and Animation of Wave Propagation in Elastic Rods

When a long thin rod is struck at one end, a hammer striking a nail for example, the impact is not felt instantaneously along the rod but rather travels through it at a fixed speed as an impact stress wave. Once it reaches the other end it will typically bounce off and travel back the way it came. The mathematics of this problem consists, essentially, of Hooke's law of elasticity and Newton's second law of motion. Once combined they produce the partial differential equation known as the 'wave equation' (see e.g. `en.wikipedia.org/wiki/Wave_equation`). In this project you will study this background theory and derive the exact solution to some representative examples. These solutions will be coded in matlab and graphical animations will be produced in order to illustrate the travelling waves. The main starting reference is Graff's book, [4] (Brunel library: QC176.8.W3G73).

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Coverage of elasticity theory and derivation of the basic equation.
- Review and development of appropriate Laplace transform theory. Derivation of exact solutions to some model problems
- Illustration of the derived solution in matlab. Graphical animation of the solution as well as derived quantities such as stress and strain. Discussion of properties of the exact solution.
- For very high marks you will have extended this work to more challenging wave propagation problems (e.g. involving dispersion) and give a deep consideration to the derivation and illustration of the solution.

Involves: theory and computing.

4.3 Theory and Computation of Impact Stresses in Elastic Tubes

What happens if all of the people in a lift jump up and land together? The sudden force felt by the lift cable is much larger than just the weight of the people. The 'impact' causes a magnification of stress. This is the reason why a hammer can drive a nail into wood, why a sudden twist can undo a stubborn bottle top and why even a low speed collision can kill and maim. The aim of this project is for the student to discover the relevant mathematics and mechanics of impact through a literature search, explain a few illustrative models and — if time allows — use software approximations to the underlying equations so as to obtain approximate solutions to realistic problems. A particular example is when a steel tube, clamped somewhere along its length, is struck suddenly at one end. What happens at the other end? What effect does the clamp size and postion have? And does how much difference does the tube's wall thickness make? To answer this means considering the problem as a two-space dimensional system of partial differential equations in polar coordinates and then employing a computer algorithm (using the finite element technique) to approximate these physics. Matlab can be used to write this code, although other choices of programming languages can be considered after proper discussion.

To get some idea of the basic ideas behind impact you can take a look at the old book by Case and Chilver (if you can find it anywhere): [2], or at Hibbeler's book *Mechanics of Materials* (Brunel library: TA405.H47 2008). These online notes `www.freestudy.co.uk/dynamics/impulse\%20and\%20momentum.pdf` also contain some useful starting material.

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Coverage of elasticity theory and derivation of the basic equations of impact.
- Cogent explanations and derivations of a set of interesting and realistic examples with, possibly, some supporting code and graphics.
- Demonstrated appreciation of how the partial differential equation known as the wave equation governs impact waves in a 1D structure.

- For very high marks you will have extended this work to more challenging wave propagation problems in a tube by using a finite element code. This will be supplied. You will use and configure it for examples, but are not required to understand how it is derived and from what.

Involves: theory and computing.

4.4 Theory and animation of pursuit problems

Pursuit problems arise in mechanics when one particle (A), travelling on a given trajectory, is pursued by another (B) in such a way that B's velocity is always toward A. Examples include a missile homing in on a plane in flight, a robot 'hand' trying to pick up a moving object, or even a dog chasing a rabbit. The mathematical model of a pursuit problem is a system of differential equations that may, or may not, be amenable to analytic solution. This project will convey the theory of pursuit problems and use matlab to arrive at numerical solutions to some example problems as well as to produce animations. For an example of what is possible see the java applet at `curvebank.calstatela.edu/pursuit2/pursuit2.htm` (and imagine how much more interesting it would be if the green particle travelled on a *curved* path between wall bounces). Two starting references are [3, 8] (Brunel library: Chorlton, QA846.C45; Smith & Smith, QA801.S63 1990) but more focussed supervisory guidance will be given once the project commences.

Broadly speaking, your final marks will reflect how far down this list of achievements you successfully go:

- Discovery of relevant literature and theory. Coverage of pursuit theory in 2D and worked solutions to some example problems. Graphical illustrations.
- Numerical solutions and animations of the pursuit curves for 2D problems where A travels a straight line.
- Extension of the above to the case where A travels a non-straight, perhaps even random, trajectory.
- For very high marks you will have extended the pursuit theory, algorithm and graphics to 3D, and you will illustrate the effect of discretisation error on the physics.

Involves: theory and computing.

References

[1] Martin Baxter and Andrew Rennie. *Financial calculus*. Cambridge University Press, 1996.

[2] John Case and A. H. Chilver. *Strength of materials and structures: an introduction to the mechanics of solids and structures*. Edward Arnold (Publishers) Ltd., second edition, 1981.

[3] F. Chorlton. *Textbook of dynamics*. Ellis Horwood, 1963.

[4] Karl F. Graff. *Wave motion in elastic solids*. Clarendon Press, Oxford, 1975.

[5] Desmond J. Higham. Nine ways to implement the binomial method for option valuation in matlab. *SIAM Review*, 44:661—677, 2002.

[6] Desmond J. Higham. *An introduction to financial option valuation; mathematics, stochastics and computation*. Cambridge University Press, New York, 2004.

[7] Nicholas J. Higham. *Handbook of writing for the mathematical sciences*. SIAM, 1998.

[8] R. C. Smith and P. Smith. *Mechanics*. Wiley, 1968.

[9] Paul Wilmott, Sam Howison, and Jeff Dewynne. *The mathematics of financial derivatives*. Cambridge University Press, 1995.