*Mathematical Tripos Part IB*

# Computational Projects
# 2025/2026

*CATAM*

# Mathematical Tripos Part IB
# Computational Projects

July 2025

*Edited by the Computational Projects Assessors Committee*

*Course Director: Prof J.R. Taylor*

*Assistant Course Director: Dr D. Frank*

**Computer-Aided Teaching of All Mathematics**

**Faculty of Mathematics**

**University of Cambridge**

# Contents



For maximum credit, you should attempt both projects from section 1 (*Core Projects* above), and then two additional projects chosen from section 2 (*Additional Projects*). You may not attempt more than two additional projects. All projects carry equal credit.

 

# Introduction

## 1 General

Please read the whole of this introductory chapter before beginning work on the projects. **It contains important information that you should know as you plan your approach to the course.**

### 1.1 Introduction

The first Computational Projects course is an element of Part IB of the Mathematical Tripos (the second Computational Projects course is an element of Part II). Although a Part IB course, lectures and introductory sessions were given as part of the Part IA year. After the lectures and sessions, and once the manual has been published, you may work at your own speed on the examinable projects.

The course is an introduction to the techniques of solving problems in mathematics using computational methods. **It is examined entirely through the submission of project reports**; there are no questions on the course in the written examination papers.

The definitive source for up-to-date information on the examination credit for the course is the Faculty of Mathematics Schedules booklet for the academic year 2025-26. At the time of writing (July 2025) the booklet for the academic year 2021-22 states that

> *No questions on the Computational Projects are set on the written examination papers, credit for examination purposes being gained by the submission of reports. The maximum credit obtainable is 160 marks and there are no alpha or beta quality marks. Credit obtained is added directly to the credit gained on the written papers. The maximum contribution to the final merit mark is thus 160, which is roughly the same (averaging over the alpha weightings) as for a 16-lecture course. Projects are considered to be a single piece of work within the Mathematical Tripos.*

### 1.2 The nature of CATAM projects

CATAM projects are intended to be exercises in independent investigation somewhat like those a mathematician might be asked to undertake in the 'real world'. They are well regarded by external examiners, employers and researchers (and you might view them as a useful item of your *curriculum vitae*).

The questions posed in the projects are more open-ended than standard Tripos questions: **there is not always a single 'correct' response**, and often the method of investigation is not fully specified. **This is deliberate.** Such an approach allows you both to demonstrate your ability to use your own judgement in such matters, and also to produce mathematically intelligent, relevant responses to imprecise questions. Particularly with respect to the Additional Projects (2.1 to 2.4), you will also gain credit for posing, and responding to, further questions of your own that are suggested by your initial observations. You are allowed and encouraged to use published literature (but it must be referenced, see also §5) to substantiate your arguments, or support your methodology.

 

## 1.3 Timetable

The timetable below is given as a guide to the expected workload.

*End of Lent Term and Easter Term, Part IA:* work through the MATLAB booklet, and attend the introductory lectures, and a MATLAB session if available. If you have no previous computing experience then you may need to spend extra time learning the basics; the summer vacation is a good opportunity to do this.

*Over the summer between Part IA and Part IB:* it is strongly recommended that you do the *optional*, non-examinable, *Introductory Project*. Unlike the other projects you may collaborate as much as you like on this project, and your College can arrange a supervision on it. A model answer will be provided towards the start of the Michaelmas Term.

Note that, if you wish, you may start the core and/or additional projects over the summer (once they are published). However,

- you are advised to attempt the *Introductory Project* first;
- please make sure that you have read and understood §5, *Unfair Means, Plagiarism and Guidelines for Collaboration*, before starting the assessed projects.

*Michaelmas Term and Christmas vacation, Part IB:* complete the programming and write-ups for the **two core projects**. A good *aim* is to finish and submit these projects by the end of the Christmas vacation.

*Lent Term and Easter vacation, Part IB:* you have one week at the start of Lent Term to make *last-minute* changes to the core projects, which should then be submitted (see §6.2 below). Then undertake **two additional projects** (out of a choice of four) and write them up.

- Between the submission deadline and the end of Lent Full Term, you may be called either for a routine *Viva Voce Examination* or, if unfair means are suspected (see §5.2 below), for an *Examination Interview* or for an *Investigative Meeting*.
- To help you assess where marks have been gained/lost, the completed marking scheme for each *Core Project* will be returned to you before the end of Lent Full Term. At that stage the Faculty will aim to offer a short "feedback" session on one of the Core Projects to those who have submitted.

*Easter Term, Part IB:* you have one week to make any *last-minute* changes to the additional projects.

*After the examinations:* **you must be available** in the last week of Easter term in case you are called either for a routine *Viva Voce Examination* or, if unfair means are suspected (see §5.2 below), for an *Examination Interview* or an *Investigative Meeting*.

### 1.3.1 Planning your work

- You are strongly advised to complete all your computing work by the end of the Christmas and Easter vacations if at all possible, since the submission deadlines are at the start of Lent and Easter Terms.
- **Do not leave writing up your projects until the last minute.** When you are writing up it is highly likely that you will either discover mistakes in your programming and/or want to refine your code. This will take time. If you wish to maximise your marks, the

 

process of programming and writing-up is likely to be **iterative**, ideally with at least a week or so between iterations.

- It is a good idea to write up each project as you go along, rather than to write all the programs first and only then to write up the reports; each year several students make this mistake and lose credit in consequence (in particular note that a program listing without a write-up, or vice versa, gains no credit). You can, indeed should, review your write-ups in the final week before the relevant submission date.

## 1.4 Programming language[s]

This year the Faculty is supporting MATLAB as the programming language. However, you are free to use any programming language that you choose as long as it does not make the programming task trivial (see discussion below). During your time in Cambridge the University will provide you with a free copy of MATLAB for your computer. Alternatively you can use the version of MATLAB that is available on the Managed Cluster Service (MCS) that is available at a number of UIS and institutional sites around the Collegiate University.

### 1.4.1 Your copy of MATLAB

All undergraduate students at the University are entitled to download and install MATLAB on their own computer that is running Windows, MacOS or Linux; your copy should be used for non-commercial University use only. The files for download, and installation instructions, are available at

`http://www.maths.cam.ac.uk/undergrad/catam/software/matlabinstall/matlab-personal.htm`.

This link is Raven protected. Several versions of MATLAB may be available; if you are downloading MATLAB for the first time it is recommended that you choose the latest version.

### 1.4.2 Programming guides and manual[s]

The Faculty of Mathematics has produced a booklet *Learning to use MATLAB for CATAM project work*, that provides a step-by-step introduction to MATLAB suitable for beginners. This is available on-line at

`http://www.maths.cam.ac.uk/undergrad/catam/MATLAB/manual/booklet.pdf`

However, this short guide can only cover a small subset of the MATLAB language. There are many other guides available on the net and in book form that cover MATLAB in far more depth. In addition:

- MATLAB has its own extensive built-in help and documentation.
- The suppliers of MATLAB, *The MathWorks*, provide MATLAB Onramp, an interactive tutorial on the basics which does not require MATLAB installation: see

  `http://uk.mathworks.com/support/learn-with-matlab-tutorials.html`

- *The MathWorks* also provide the introductory guide *Getting Started with MATLAB*. You can access this by 'left-clicking' on the `Getting Started` link at the top of a MATLAB *'Command Window'*. Alternatively there is an on-line version available at

  `http://uk.mathworks.com/help/matlab/getting-started-with-matlab.html`

 

- Further, *The MathWorks* provide links to a whole a raft of other tutorials; see

  https://uk.mathworks.com/support/learn-with-matlab-tutorials.html

  In addition their *Matlab* documentation page gives more details on maths, graphics, object-oriented programming etc.; see

  http://uk.mathworks.com/help/matlab/index.html

- There is a plethora of books on Matlab. For instance:

  (a) *Numerical Computing with Matlab* by Cleve Moler (SIAM, Second Edition, 2008, ISBN 978-0-898716-60-3). This book can be downloaded for free from

  http://uk.mathworks.com/moler/chapters.html

  (b) *Matlab Guide* by D.J. Higham & N.J. Higham (SIAM, Second Edition, 2005, ISBN 0-89871-578-4).

  You may be spoilt for choice: Google returns about 100,000,000 hits for the search 'Matlab introduction', and about 11,000,000 hits for the search 'Matlab introduction tutorial'.

- The Engineering Department has a webpage that lists a number of helpful articles; see

  http://www.eng.cam.ac.uk/help/tpl/programs/matlab.html

Similarly, many online resources are available for learning other programming languages (e.g. python, R, Julia, etc.)

#### 1.4.3 To Matlab, or not to Matlab

Use of Matlab is recommended,[1] especially if you have not programmed before, but *you are free to write your programs in any computing language whatsoever.* Python, Julia,[2] R,[3] C, C++, Mathematica,[4] Maple[5] and Haskell have been used by several students in the past, and Excel has been used for plotting graphs of computed results. The choice is your own, provided your system can produce results and program listings for inclusion in your report.[6]

However, you should bear in mind the following points.

- The Faculty does *not* promise to help you with programming problems if you use a language other than Matlab.

[1] Except where an alternative is explicitly stated, e.g. see footnotes 3 and 5.

[2] Julia is a high-level open source language well suited to numerical computation. An Introduction to Julia for CATAM under ongoing development is available from https://sje30.github.io/catam-julia/.

[3] R is a programming language and software environment for statistical and numerical computing, as well as visualisation. It is the recommended language for some Part II projects. R is available for free download for the Linux, MacOS and Windows operating systems from http://www.r-project.org/.

[4] Mathematica is a software package that supports symbolic computations and arbitrary precision numerical calculations, as well as visualisation. At the time of writing Mathematica is also available for free to mathematics students, but the agreement is subject to renewal. You can download versions of Mathematica for the Linux, MacOS and Windows operating systems from

https://www.maths.cam.ac.uk/computing/software/mathematica/

[5] Maple is a mathematics software package that supports symbolic computations and arbitrary precision numerical calculations, as well as visualisation. It is the recommended language for some Part II projects.

[6] There is no need to consult the *CATAM Helpline* as to your choice of language.

 

- Not all languages have the breadth of mathematical routines that come with the MATLAB package. You may discover either that you have to find reliable replacements, or that you have to write your own versions of mathematical library routines that are pre-supplied in MATLAB (this can involve a fair amount of effort). To this end you may find reference books, such as *Numerical Recipes* by W. H. Press *et al.* (CUP), useful. You may use equivalent routines to those in MATLAB from such works so long as you acknowledge them, and reference them, in your write-ups.

- If you choose a high-level programming language that can perform advanced mathematical operations automatically, then you should check whether use of such commands is permitted in a particular project. As a rule of thumb, do not use a built-in function if there is no equivalent MATLAB routine that has been approved for use in the project description, or if use of the built-in function would make the programming considerably easier than intended. For example, use of a command to test whether an integer is prime would not be allowed in a project which required you to write a program to find prime numbers. The *CATAM Helpline* (see §4 below) can give clarification in specific cases.

- Subject to the aforementioned limited exceptions, *you must write your own computer programs*. Downloading computer code, e.g. from the internet, that you are asked to write yourself counts as plagiarism (see §5).

## 2 Project Reports

### 2.1 Project write-ups: examination credit

Each individual project carries the same credit. For each project, 40% of the marks available are awarded for writing programs that work and for producing correct graphs, tables of results and so on. A further 50% of the marks are awarded for answering mathematical questions in the project and for making appropriate mathematical observations about your results.

The final 10% of marks are awarded for the overall 'excellence' of the write-up. Half of these 'excellence' marks may be awarded for presentation, that is for producing good clear output (graphs, tables, etc.) which is easy to understand and interpret, and for the mathematical clarity of your report.

The assessors may penalise a write-up that contains an excessive quantity of irrelevant material (see below). In such cases, the 'excellence' mark may be reduced and could even become negative, as low as -10%.

Unless the project specifies a way in which an algorithm should be implemented, marks are, in general, not awarded for programming style, good or bad. Conversely, if your output is poorly presented — for example, if your graphs are too small to be readable or are not annotated — then you may lose marks.

| |
|---|
| **No marks** are given for the submission of program code without a report, or vice versa. |

The marks for each project are scaled so that a possible maximum of 160 marks are available for the Part IB Computational Projects course. No quality marks (i.e. $\alpha$s or $\beta$s) are awarded. The maximum contribution to the final merit mark is thus 160 and roughly the same (averaging over the $\alpha$ weightings) as for a 16-lecture course.

## 2.2 Project write-ups: advice

Your record of the work done on each project should contain all the results asked for and your comments on these results, together with any graphs or tables asked for, clearly labelled and referred to in the report. However, it is important to remember that the project is set as a piece of mathematics, rather than an exercise in computer programming; thus the most important aspect of the write-up is the **mathematical content**. For instance:

- Your comments on the results of the programs should go *beyond* a rehearsal of the program output and show an understanding of the mathematical and, if relevant, physical points involved. The write-up should demonstrate that you have noticed the most important features of your results, and understood the relevant mathematical background.

- When discussing the computational method you have used, you should distinguish between points of interest in the algorithm itself, and details of your own particular implementation. Discussion of the latter is usually unnecessary, but if there is some reason for including it, please set it aside in your report under a special heading: it is rare for the assessors to be interested in the details of how your programs work.

- Your comments should be pertinent and concise. Brief notes are perfectly satisfactory — provided that you cover the salient points, and make your meaning precise and unambiguous — indeed, students who keep their comments concise can get better marks. An over-long report may well lead an assessor to the conclusion that the candidate is unsure of the essentials of a project and is using quantity in an attempt to hide the lack of quality. Do not copy out chunks of the text of the projects themselves: you may assume that the assessor is familiar with the background to each project and all the relevant equations.

- Similarly you should not reproduce large chunks of your lecture notes; you will not gain credit for doing so (and indeed may lose credit as detailed in §2.1). However, you will be expected to reference results from theory, and show that you understand how they relate to your results. If you quote a theoretical result from a textbook, or from your notes, or from the WWW, you should give both a brief justification of the result and a *full reference*.[7] If you are actually asked to *prove* a result, you should do so concisely.

- Graphs will sometimes be required, for instance to reveal some qualitative features of your results. Such graphs, *including labels, annotations, etc.*, need to be computer-generated (see also § 2.3). Further, while it may be easier to include only one graph per page, it is often desirable (e.g. to aid comparison) to include *two or more* graphs on a page. Also, do not forget to clearly label the axes of graphs or other plots, and provide any other annotation necessary to interpret what is displayed. Similarly, the rows and columns of any tables produced should be clearly labelled.

- You should take care to ensure that the assessor sees evidence that **your programs do indeed perform the tasks** you claim they do. In most cases, this can be achieved by including a sample output from the program. If a question asks you to write a program to perform a task but doesn't specify explicitly that you should use it on any particular data, you should provide some 'test' data to run it on and include sample output in your write-up. Similarly, if a project asks you to 'print' or 'display' a numerical result, you should demonstrate that your program does indeed do this by including the output.

[7] See also the paragraph on *Citations* in §5

 

- **Above all, make sure you comment where the manual specifically asks you to.** It also helps the assessors if you answer the questions in the order that they appear in the manual and, if applicable, **number your answers** using the same numbering scheme as that used by the project. Make clear which outputs, tables and graphs correspond to which questions and programs.

The following are indicative of some points that might be addressed in the report; they are not exhaustive and, of course not all will be appropriate for every project. In particular, some are more relevant to pure mathematical projects, and others to applied ones.

- Does the algorithm or method always work? Have you tested it?
- What is the theoretical running time, or complexity, of the algorithm? Note that this should be measured by the number of simple operations required, expressed in the usual $O\big(f(n)\big)$ or $\Omega\big(f(n)\big)$ notation, where $n$ is some reasonable measure of the size of the input (say the number of vertices of a graph) and $f$ is a reasonably simple function. Examples of simple operations are the addition or multiplication of two numbers, or the checking of the $(p, q)$ entry of a matrix to see if it is non-zero; with this definition finding the scalar product of two vectors of length $n$ takes order $n$ operations. Note that this measure of complexity can differ from the number of MATLAB commands/'operations', e.g. there is a single MATLAB command to find a scalar product of two vectors of length $n$.
- What is the accuracy of the numerical method? Is it particularly appropriate for the problem in question and, if so, why? How did you choose the step-size (if relevant), and how did you confirm that your numerical results are reliably accurate for all calculations performed?
- How do the numerical answers you obtain relate to the mathematical or physical system being modelled? What conjectures or conclusions, if any, can you make from your results about the physical system or abstract mathematical object under consideration?

In summary, it is the candidate's responsibility to determine which points require discussion in the report, to address these points fully but concisely, and to structure the whole so as to present a clear and complete response to the project. It should be possible to read your write-up without reference to the listing of your programs.

As an aid, for the **two core projects** only, some brief additional comments are provided giving further guidance as to the form and approximate length of answer expected for each question. These also contain a mark-scheme, on which your marks for each question will be written and returned to you during the Lent Term. For the additional projects you are expected to use your judgement on the marks allocation.

### 2.2.1 Project write-ups: advice on length

The word *brief* peppers the last few paragraphs. To emphasise this point, in general **six sides of A4 of text**, *excluding in-line graphs, tables, etc.*, should be plenty for a clear concise report. Indeed, the best reports are sometimes shorter than this.

To this total you will of course need to add tables, graphs etc. However, *do **not** include every single piece of output you generate*: include a selection of the output that is a *representative* sample of graphs and tables. It is up to you to choose a selection which demonstrates all the important features but is reasonably concise. Presenting mathematical results in a clear and concise way is an important skill and one that you will be evaluated upon in CATAM. Twenty pages of graphs would be excessive for most projects, even if the graphs were one to a page.[8] Remember that the assessors will be allowed to **deduct** up to 10% of marks for any project containing an excessive quantity of irrelevant material. Typically, such a project might be long-winded, be very poorly structured, or contain long sections of prose that are not pertinent. Moreover, if your answer to the question posed is buried within a lot of irrelevant material then it may not receive credit, even if it is correct.

### 2.3 Project write-ups: technicalities

As emphasised above, elaborate write-ups are not required. You are required to submit your project reports electronically. In particular, you will be asked to submit your write-ups electronically in Portable Document Format (PDF) form, and you should ensure that the submitted file can be printed (in portrait mode on standard A4 paper). Note that many word processors (e.g. LaTeX, *Microsoft Word*, *LibreOffice*) will generate output in PDF form. In addition, there are utility programs to convert output from one form to another, in particular to PDF form (e.g. there are programs that will convert plain text to PDF). Before you make your choice of word processor, you should confirm that you will be able to generate submittable output in PDF form. Please note that a PDF file including pages generated by scanning a hand-written report or other text document is ***not*** acceptable.

In a very few projects, where a *sketch* (or similar) is asked for, a scanned hand-drawing is acceptable. Such exceptions will be noted *explicitly* in the project description.

If it will prove difficult for you to produce electronic write-ups, e.g. because of a disability, then please contact the *CATAM Helpline* as early as possible in the academic year, so that reasonable adjustments can be made for you.

*Choice of Word Processor.* As to the choice of word processor, there is no definitive answer. Many mathematicians use LaTeX (or, if they are of an older generation, TeX), e.g. this document is written in LaTeX. However, please note that although LaTeX is well suited for mathematical typesetting, it is absolutely acceptable to write reports using other word-processing software, e.g. *Microsoft Word* or *LibreOffice*.

- *Microsoft Word* is commercial, but is available free while you are a student at Cambridge: see

  `https://help.uis.cam.ac.uk/service/collaboration/office365`.

- *LibreOffice* can be installed for free for, *inter alia*, the Windows, MacOS and Linux operating systems from

  `http://www.libreoffice.org/download/download/`.

*LaTeX.* If you decide to use LaTeX, you can use an online editor or install LaTeXon your own personal computer.

If you use an online editor, *it is essential that you do not share your project with anyone* (see section on Unfair Means). A popular online LaTeXeditor is called Overleaf. The University currently provides a 'Professional' level Overleaf license for students, see

`https://www.overleaf.com/edu/cambridge`.

If you use Overleaf or another online editor, we strongly advise keeping a local copy of your report files.

[8] Recall that graphs should not as a rule be printed one to a page.

Whether you use an online editor or not, you will probably want to install it on your own personal computer. This can be done for free. For recommendations of TEX distributions and associated packages see

- http://www.tug.org/begin.html and
- http://www.tug.org/interest.html.

*Front end.* In addition to a TEX distribution you will also need a front-end (i.e. a 'clever editor'). A comparison of TEX editors is available on WIKIPEDIA; below we list a few of the more popular TEX editors.

*TEXstudio.* For Windows, Mac and Linux users, there is TEXstudio. The proTEXt distribution, based on MiKTEX, includes the TEXstudio front end.

*TEXworks.* Again for Windows, Mac and Linux users, there is TEXworks. The MiKTEX distribution includes TEXworks.

*TEXShop.* Many Mac aficionados use TEXShop. To obtain TEXShop and the TEXLive distribution see http://pages.uoregon.edu/koch/texshop/obtaining.html.

*TEXnicCenter.* TEXnicCenter is another [older] front end for Windows users.

*Learning LATEX.* A *Brief LATEX Guide for CATAM* is available for download from

http://www.maths.cam.ac.uk/undergrad/catam/files/Brief-Guide.pdf .

- The LATEX source file (which may be helpful as a template), and supporting files, are available for download as a zip file from

  http://www.maths.cam.ac.uk/undergrad/catam/files/Guide.zip .

  Mac, Unix and most Windows users should already have an unzip utility. Windows users can download 7-Zip if they have not.

*Layout of the first page.* The first page of your report should include the **project name** and **project number**.

*Your script is marked anonymously.* Hence, your **name or user identifier should not appear anywhere** in the write-up (including any output).

*Further technicalities.* Please do not use red or green for text (although red and/or green lines on plots are acceptable). Please leave a margin at least 2 cm wide at the left, and number each page, table and graph.

*Program listings.* At the end of each report you should include complete **listings** (i.e. printout of source code) of every major program used to generate your results. You do *not* need to include a listing of a program which is essentially a minor revision of another which you have already included. Make sure that your program listings are the *very last* thing in your reports. Please do not mix program output and program listings together; if you do, the program output may not be marked as part of the report.

## 3 Computing Facilities

You may write and run your programs on any computer you wish, whether it belongs to you personally, to your College, or to the University. If you believe that do not have access to an adequate computer to complete the CATAM projects, you should contact your Director of Studies and/or the CATAM helpline *well in advance* of any project deadlines.

### 3.1 Backups

Whatever computing facilities you use, **make sure you make regular backups of your work** in case of disaster! Remember that occasionally systems go down or disks crash or computers are stolen. **Malfunctions of your own equipment or the MCS are not an excuse for late submissions**: leave yourself enough time before the deadline.

Possibly one of the easiest ways to ensure that your work is backed up is to use an online 'cloud' service; many of these services offer some free space. WIKIPEDIA has a fairly comprehensive list at http://en.wikipedia.org/wiki/Comparison_of_online_backup_services. In particular note that eligible students have 1TB of OneDrive personal storage space via their University Microsoft account and 500Gb of personal storage on Google Drive under a University agreement (see https://help.uis.cam.ac.uk/service/data-and-file-storage/store-documents-are-personal-you).

## 4 Information Sources

There are many ways of getting help on matters relating to CATAM.

*The CATAM Web Page.* The CATAM web page,

http://www.maths.cam.ac.uk/undergrad/catam/

contains much useful information relating to CATAM. There are on-line, and up-to-date, copies of the projects, and any data files required by the projects can be downloaded. There is also the booklet *Learning to use MATLAB for CATAM project work*.

*CATAM News and Email.* Any important information about CATAM (e.g. corrections to projects or to other information in the *Manual*) is publicised via *CATAM News*, which can be reached from the CATAM web page. You must read *CATAM News* from time to time (e.g. just before starting a project) to check for these and other important announcements, such as submission dates and procedures.

As well as adding announcements to *CATAM News*, occasionally we will email students using the year lists maintained by the Faculty of Mathematics. You have a responsibility to read email from the Faculty, and if we send an email to one of those lists we will assume that you have read it.

After 1 October 2025 you can check that you are on the appropriate Faculty year list by referring to the https://lists.cam.ac.uk/mailman/raven webpage (to view this page you will need to authenticate using Raven if you have not already done so). You should check that the *Maths-IB* mailing list is one of your current lists.

If you are not subscribed to the correct mailing list, then this can be corrected by contacting the Faculty Undergraduate Office (email: undergrad-office@maths.cam.ac.uk) with a request to be subscribed to the correct list (and, if necessary, unsubscribed from the wrong list).

*The CATAM Helpline.* If you need help (e.g. if you need clarification about the wording of a project, or if you have queries about programming and/or MATLAB), you can email a query to the *CATAM Helpline*: catam@maths.cam.ac.uk. Almost all queries may be sent to the *Helpline*, and it is particularly useful to report potential errors in projects. However the *Helpline* cannot answer detailed mathematical questions about particular projects. Indeed if your query directly addresses a question in a project you may receive a standard reply indicating that the *Helpline* cannot add anything more.

In order to help us manage the emails that we receive,

- please use an email address ending in `cam.ac.uk` (rather than a Gmail, etc. address) both so that we may identify you and also so that your email is not identified as spam;
- please specify, in the subject line of your email, 'Part IB' as well as the project number and title or other topic, such as 'MATLAB query', to which your email relates;
- please also **restrict each email to one question or comment** (use multiple emails if you have more than one question or comment).

The *Helpline* is available during Full Term and one week either side. Queries sent outside these dates will be answered subject to personnel availability. We will endeavour (but do not guarantee) to provide a response from the *Helpline* within three working days. However, if the query has to be referred to an assessor, then it may take longer to receive a reply. Please do not send emails to any other address.

*The CATAM FAQ Web Pages.* Before asking the *Helpline* about a particular project, please check the *CATAM FAQ web pages* (accessible from the main CATAM web page). These list questions which students regularly ask, and you may find that your query has already been addressed.

*Advice from Supervisors and Directors of Studies.* The general rule is that advice **must be general in nature**. You should not have supervisions on any work that is yet to be submitted for examination; however, you may have a supervision on the *Introductory Project*, and/or another non-examinable project, and/or any work set by your Director of Studies. A supervisor can also provide feedback on the *Core Projects* **after** they have been submitted (e.g. **after** your marks have been returned). To that end the Faculty will aim to offer you a short "feedback" session on one of the *Core Projects* towards the end of the Lent Term.

## 5 Unfair Means, Plagiarism and Guidelines for Collaboration

The objective of CATAM is for you to learn computational methods, mathematics and written presentation skills. To achieve these objectives, you must work independently on the projects, both on the programming and on the write-ups.

The work that you turn in **must** be your own. This applies equally to the source code and the write-ups, i.e. you must write and test all programs yourself, and all reports must be written *independently*.

Any attempt to gain an unfair advantage, for example by copying computer code, mathematics, or written text, is not acceptable and will be subject to serious sanctions.

If you have any questions about what constitutes unfair means, you should seek advice from the CATAM helpline.

*Citations.* It is, of course, perfectly permissible to use reference books, journals, reference articles on the WWW or other similar material: indeed, you are encouraged to do this. You may quote directly from reference works so long as you acknowledge the source (WWW pages should be acknowledged by a *full* URL). There is no need to quote lengthy proofs in full, but you should at least include your own brief summary of the material, together with a *full* reference (including, if appropriate, the page number) of the proof.

*Programs.* You must write your own computer programs. Downloading computer code, e.g. from the internet, that you are asked to write yourself counts as plagiarism even if cited.

*Acceptable collaboration.* It is recognised that some candidates may occasionally wish to discuss their work with others doing similar projects. This can be educationally beneficial and is accepted provided that it remains within reasonable bounds. Acceptable collaboration may include an *occasional general discussion* of the approach to a project and of the numerical algorithms needed to solve it. Small hints on debugging code (note the *small*), as might be provided by an adviser, are also acceptable.

*Unacceptable collaboration (also known as collusion).* If a general discussion *either* is happening regularly *or* gets to the point where physical or virtual notes are being exchanged (even on the back of an envelope, napkin or stamp), then it has reached the stage of unacceptable collaboration. As an example to clarify the limits of 'acceptable collaboration', if an assessor reading two anonymous write-ups were to see significant similarities in results, answers, mathematical approach or programming which would clearly not be expected from students working independently, then there would appear to be a case that the students have breached the limits. An *Investigative Meeting* would then be arranged (unless such similarities were deemed to be justified in light of the declared lists of discussions, see below). If you are uncertain about what constitutes an *unacceptable collaboration* you should seek advice from the CATAM Helpline.

*Generative AI.* Using generative AI (e.g. ChatGPT, Gemini, Claude and similar) to produce some or part of the submitted write-up or source code would not be original work and hence is considered a form of academic misconduct. This interpretation is consistent with *University guidelines*. We use software that is capable of detecting AI-generated content, and where a case of unfair means is suspected, the Examiners may, at their discretion, examine a candidate by means of an Oral Examination.

The following actions are examples of *unfair means*

- copying any other person's program, either automatically or by typing it in from a listing;
- using someone else's program or any part of it as a model, or working from a jointly produced detailed program outline;
- copying or paraphrasing of someone else's report in whole or in part;
- turning in output from a generative AI either in the report or in the source code.

These comments apply just as much to copying from the work of previous Part IB students, or another third party (including any code, etc. you find on the internet), as they do to copying from the work of students in your own year. Asking anyone for help that goes past the limits of *acceptable collaboration* as outlined above, and this includes posting questions on the internet (e.g. StackExchange), constitutes *unfair means.*

Further, you should not allow any present or future Part IB student access to the work you have undertaken for your own CATAM projects, even after you have submitted your write-ups. If you knowingly give another student access to your CATAM work you are in breach of these guidelines and may be charged with assisting another candidate to make use of unfair means.

### 5.1 Further information about policies regarding plagiarism and other forms of unfair means

*University-wide Statement on Plagiarism.* You should familiarise yourself with the University's *Statement on Plagiarism*.

There is a link to this statement from the University's *Good academic practice and plagiarism* website

http://www.plagiarism.admin.cam.ac.uk/,

which also features links to other useful resources, information and guidance.

*Faculty Guidelines on Plagiarism.* You should also be familiar with the Faculty of Mathematics Guidelines on Plagiarism. These guidelines, which include advice on quoting, paraphrasing, referencing, general indebtedness, and the use of web sources, are posted on the Faculty's website at

http://www.maths.cam.ac.uk/facultyboard/plagiarism/.

In order to preserve the academic integrity of the Computational Projects component of the Mathematical Tripos, the following procedures have been adopted.

*Declarations.* To certify that you have *read and understood* these guidelines, you will be asked to sign an electronic declaration. Further instructions will be given during Michaelmas Term.

In order to certify that you have *observed* these guidelines, you will be required to sign an electronic submission form provided when you submit your write-ups, and you are advised to read it carefully; it will be similar to that reproduced (subject to revision) as Appendix A. You must list on the form *anybody* (students, supervisors and Directors of Studies alike) with whom you have exchanged information (e.g. by talking to them, or by electronic means) about the projects at any more than a *trivial* level: *any* discussions that affected your approach to the projects to *any* extent must be listed. Failure to include on your submission form any discussion you may have had *is* a breach of these guidelines.

However, declared exchanges are perfectly allowable so long as they fall within the limits of 'acceptable collaboration' as defined above, and you should feel no qualms about listing them. For instance, as long as you have refrained from discussing in any detail your programs or write-ups with others after starting work on them, then the limits have probably not been breached.

The assessors will not have knowledge of your declaration until after all your projects have been marked. However, your declaration may affect your CATAM marks if the assessors believe that discussions have gone beyond the limits of what is acceptable. If so, or if there is a suspicion that your have breached any of the other guidelines, you will be summoned to an *Investigative Meeting* (see §5.2). Ultimately, your case could be brought to the University courts and serious penalties could result (see *Sanctions* below).

*Plagiarism detection.* **The programs and reports submitted will be checked carefully both to ensure that they are your own work, and to ensure the results that you hand in have been produced by your own programs.**

*Checks on submitted program code.* The Faculty of Mathematics uses (and has used for many years) specialised software, including that of external service providers, which automatically checks whether your programs either have been copied or have unacceptable overlaps (e.g. the software can spot changes of notation). All programs submitted are screened.

The code that you submit, and the code that your predecessors submitted, is kept in *anonymised* form to check against code submitted in subsequent years.

*Checks on electronically submitted reports.* In addition, the Faculty of Mathematics will screen your electronically submitted reports using the *Turnitin UK* text-matching software. Further information will be sent to you before the submission date. The electronic declaration which you will be asked to complete at the start of the Michaelmas term will, *inter alia*, cover the use of *Turnitin UK*.

Your electronically submitted write-ups will be kept in *anonymised* form to check against write-ups submitted in subsequent years.

*Sanctions.* If plagiarism, collusion or any other method of unfair means is suspected in the Computational Projects, normally the Chair of Examiners will convene an *Investigative Meeting* (see §5.2). If the Chair of Examiners deems that unfair means were used, the case may be brought to the University courts. According to the Statues and Ordinances of the University [9]

> suspected cases of the use of unfair means (of which plagiarism is one form) will be investigated and may be brought to one of the University courts or disciplinary panels. The University courts and disciplinary panels have wide powers to discipline those found to have used unfair means in an examination, including depriving such persons of membership of the University, and deprivation of a degree.

**The Faculty of Mathematics wishes to make it clear that any breach of these guidelines will be treated very seriously.**

However, we also wish to emphasise that the great majority of candidates have, in the past, had no difficulty in keeping to these guidelines. Unfortunately there have been a small number of cases in recent years where *some individuals have been penalised by the loss of significant numbers of marks, indeed sufficient to drop a class.* If you find the guidelines unclear in any way you should seek advice from the *CATAM Helpline*. These policies and practices have been put in to place so that you can be sure that the hard work you put into CATAM will be fairly rewarded.

## 5.2 Oral examinations

*Viva Voce Examinations.* A number of candidates may be selected, either randomly or formulaically, for a *Viva Voce Examination* after submission of either the core or the additional projects. This is a matter of routine, and therefore a summons to a *Viva Voce Examination* should not be taken to indicate that there is anything amiss. You will be asked some straightforward questions on your project work, and may be asked to elaborate on the extent of discussions you may have had with other students. So long as you can demonstrate that your write-ups are indeed your own, your answers will not alter your project marks.

[9] From `https://www.admin.cam.ac.uk/univ/so/`.

 

*Examination Interviews.* Additionally, the Chair of Examiners may summon a particular candidate or particular candidates for interview on any aspect of the written work of the candidate or candidates not produced in an examination room which in the opinion of the Examiners requires elucidation. If plagiarism or other unfair means is suspected, an Investigative Meeting will be convened (see below).

*Investigative Meetings.* When plagiarism, collusion or other unfair means are suspected the Chair of Examiners may summon a candidate to an *Investigative Meeting*. If this happens, you have the right to be accompanied by your Tutor (or another representative at your request). The reasons for the meeting, together with copies of supporting evidence and other relevant documentation, will be given to your Tutor (or other representative). One possible outcome is that the case is brought to the University courts where *serious penalties can be imposed* (see *Sanctions* above).

*Timing.* *Viva Voce Examinations*, *Examination Interviews* and *Investigative Meetings* are a formal part of the Tripos examination, and if you are summoned then you must attend. In the case of the core projects these will usually take place during Lent Full Term (although they may take place exceptionally during Easter Full Term), and in the case of the additional projects these will usually take place during the last week of Easter Full Term. Easter Term *Viva Voce Examinations* are likely to take place on the Monday of the last week (i.e. Monday 15th June 2025), while *Examination Interviews* and *Investigative Meetings* may take place any time that week. If you need to attend during the last week of Easter Full Term you will be informed in writing just after the end of the written examinations. **You must be available** in the last week of Easter Full Term in case you are summoned.

# 6 Submission and Assessment

In order to gain examination credit for the work that you do on this course, you must write reports on each of the projects that you have done. As emphasised earlier it is the quality (not quantity) of your written report which is the most important factor in determining the marks that you will be awarded.

## 6.1 Submission form

When you submit your project reports you will be required to complete and upload the submission form provided, detailing which projects you have attempted and listing all discussions you have had concerning CATAM (see §5, *Unfair Means, Plagiarism and Guidelines for Collaboration*, and Appendix A). Further details, including the definitive submission form, will be made available when the arrangements for electronic submission of reports and programs (see below) are announced.

## 6.2 Submission of written work

In order to gain examination credit, you must:

- submit electronic copies of your reports and programs (see §6.3);
- complete and submit your submission form listing each project for which you wish to gain credit.

 

Further details about submission arrangements will be announced via *CATAM News* and email closer to the time.

For the core projects the submission deadline is

**Wednesday 21st January 2026, 4pm**,

while for the additional projects the submission deadline is

**Wednesday 29th April 2026, 4pm**.

Self-certified extensions may be obtained for a period of up to 7 days. A form to apply for an extension will be made available on Moodle at the start of the submission period. Students must inform their college Tutor that they are applying for an extension *before* filling out the extension request form. Whenever possible, students should apply for an extension before the original submission deadline. It will not be possible to apply for a self-certified extension later than 7 days after the original submission deadline. We strongly encourage all students to complete and turn-in their work well in advance of the original submission deadline to allow time to deal with any issues arising during submission and avoid impacts on other coursework and revision.

The Computational Projects Assessors Committee reserves the right to **reduce** the marks awarded for any projects (including reports and source code) which are submitted late (either the standard or extended deadline, as appropriate).

### 6.3 Electronic submission

You will be required to submit electronically copies of both your reports and your program source files. Electronic submission enables the Faculty to run automatic checks on the independence of your work, and also allows your programs to be inspected in depth (and if necessary run) by the assessors.

As regards your programs, electronic submission applies whether you have done your work on your own computer, on the MCS, or elsewhere, and is regardless of which programming language you have chosen.

Details of the procedure will be given in advance of the submission deadlines via *CATAM News* and email.

**However please note that you will need to know your UIS password in order to submit copies of your report and program source files.**

If you cannot remember your UIS password you will need to follow that instructions provided by the University Information Service.[10] Note that if you need a Password Reset Token then this may take some time to obtain, so check that you know your UIS password well before submission day.

### 6.4 Saving and sharing electronic files

After the submission deadline the electronic files will be taken offline and you will not be able to download your submitted work from the submission site. We recommend that you keep electronic copies of your work.

[10] See `https://password.raven.cam.ac.uk/`.

A copy of your submission is likely to be particularly useful for the core projects for which you will be given a breakdown of the marks you have obtained. Since the manuals will be taken off-line after the close of submission, you might also like to save a copy of the projects you have attempted.

**It is critical that you do not make your reports or source code available to any present or future students. This includes posting to publically accessible repositories such as github.**

Please note that all material that you submit electronically is kept in *anonymised* form to check against write-ups and program code submitted in subsequent years.

### 6.5 Returning from intermission

If a student is returning from intermission that began in an academic year during which they submitted some or all of the CATAM projects, then in certain circumstances it is possible to carry forward some or all of their CATAM marks from that year. Action is required by the Director of Studies. Hence, before attempting any further CATAM work, the student should discuss the options available with their Director of Studies and decide on their intended strategy.

The following general policies have been approved by the Faculty Board. If there are exceptional circumstances in which these seem inappropriate, the Director of Studies should discuss these with the CATAM Director: catam-director@maths.cam.ac.uk.

If a Part IB student intermits, and then repeats the entire year starting in Michaelmas Term, they should normally be expected to start CATAM afresh as a logical part of repeating the year.

If the student returns in the Lent Term, then they may EITHER carry forward any marks earned on the core projects in a previous year OR instead submit work for the current year's core projects. The student (following discussion with the DoS) must choose which option to take before the Lent Term submission date, and submission of new work will be taken to mean they have (irrevocably) taken the latter option. If they opt, instead, to carry their marks forward, then their Director of Studies must notify the Undergraduate Office and the CATAM Director that this should be done. The student would normally be expected to complete the current year's additional projects as a logical part of repeating the Lent Term.

If the student returns in Easter Term, then any marks earned on the core projects in a previous year should be carried forward. The student may EITHER carry forward any marks earned on the additional projects in a previous year OR instead submit work for the current year's additional projects. They must choose which option to take before the Easter Term submission date, and submission of new work will be taken to mean they have (irrevocably) taken the latter option. The Director of Studies must notify the Undergraduate Office and the CATAM Director about any marks that are to be carried forward.

 

# A Appendix: Example Submission Form

**PART IB MATHEMATICAL TRIPOS 2025-26**

**Computational Projects 2025**

**CORE COMPUTATIONAL PROJECTS**

**STATEMENT OF PROJECTS SUBMITTED FOR EXAMINATION CREDIT**

**Please observe these points when submitting your CATAM projects:**

1. Your name, College or CRSid User Identifier **must not** appear anywhere in the submitted work.
2. Complete this declaration form and submit it electronically with your reports.
3. The Moodle submission site will close at 4pm on submission day and it is likely to be slow immediately prior to the deadline. Please turn in your work earlier if possible and be prepared for delays in the website on submission day.

**IMPORTANT**

Candidates are reminded that Discipline Regulation 7 reads:

> No candidate shall make use of unfair means in any University examination. Unfair means shall include plagiarism[11] and, unless such possession is specifically authorized, the possession of any book, paper or other material relevant to the examination. No member of the University shall assist a candidate to make use of such unfair means.

To confirm that you are aware of this, you **must** check and sign the declaration below and include it with your work when it is submitted for credit.

> *The Faculty of Mathematics wishes to make it clear that failure to comply with this requirement is a serious matter that could render you liable to sanctions imposed by the University courts.*

[11] Plagiarism is defined as submitting as one's own work, irrespective of intent to deceive, that which derives in part or in its entirety from the work of others without due acknowledgement.

**DECLARATION BY CANDIDATE**

I hereby submit my reports on the following projects and wish them to be assessed for examination credit:

| Project Number | Brief Title |
|---|---|
| | |
| | |

I certify that I have read and understood the section *Unfair Means, Plagiarism and Guidelines for Collaboration* in the Projects Manual (including the references therein), and that I have conformed with the guidelines given there as regards any work submitted for assessment at the University. I understand that the penalties may be severe if I am found to have not kept to the guidelines in the section *Unfair Means, Plagiarism and Guidelines for Collaboration.* I agree to the Faculty of Mathematics using specialised software, including *Turnitin UK*, to automatically check whether my submitted work has been copied or plagiarised and, in particular, I certify that

- the composing and writing of these project reports is my own unaided work and no part of it is a copy or paraphrase of work of anyone other than myself;
- the computer programs and listings and results were not copied from anyone or from anywhere (apart from the course material provided);
- I have not shown my programs or written work to any other candidate or allowed anyone else to have access to them;
- I have listed below anybody, other than the CATAM Helpline or CATAM advisers, with whom I have had discussions or exchanged information at any more than a trivial level about the CATAM projects, together with the nature of those discussions and/or exchanges.

**Declaration of Discussions and Exchanges (continue on a separate sheet if necessary)**

Signed ............................................... Date ..................................

 

# 0.1 Root Finding in One Dimension

*This is an optional, introductory, non-examinable project. Unlike the other projects* ***there are no marks awarded for it****. Also, unlike the other projects, you may collaborate as much as you like, and (if your College is willing) have a supervision on the project. A model answer will be provided on the CATAM web site towards the start of the Michaelmas Term.*

## The Methods

The aim of this project is to study iteration methods for the numerical solution of an algebraic or transcendental equation $F(x) = 0$. We consider two methods.

(i) *Binary search* (also known as bisection or interval halving).

(ii) *Fixed-point iteration*, which involves solving an equivalent system $x = f(x)$ by use of an iteration scheme

$$x_N = f\left(x_{N-1}\right) , \tag{1}$$

with a suitable initial guess $x_0$. We will consider two cases of fixed-point iteration:

(a) first, we will study an equivalent system derived by manipulating $F(x) = 0$ algebraically to the (non-unique) form $x = f(x)$;

(b) second, we will study *Newton-Raphson iteration*, which uses the scheme

$$x_N = x_{N-1} - \frac{F\left(x_{N-1}\right)}{F'\left(x_{N-1}\right)} . \tag{2}$$

The theoretical background to these methods is covered in most textbooks on *Numerical Analysis* (a few of which are listed at the end of this project).

## Order of Convergence

A sequence $\{\delta_N\}$ which converges to zero as $N \to \infty$ is said, for the purposes of this project,[1] to have order of convergence $p$ ($\geqslant 1$) if

$$|\delta_N| \sim C|\delta_{N-1}|^p \text{ as } N \to \infty , \quad \text{i.e.} \quad \lim_{N\to\infty} \frac{|\delta_N|}{|\delta_{N-1}|^p} = C , \tag{3}$$

where $C$ is some strictly positive (finite) constant; first-order (or 'linear') convergence, $p = 1$, requires $C < 1$.

If an iteration method is attempting to approximate the exact root $x_*$, the *truncation error* in the $N^{th}$ iterate is defined as $\epsilon_N = x_N - x_*$.[2] If the method is convergent, i.e. $\epsilon_N \to 0$ as $N \to \infty$, it is said to be $p^{th}$-order convergent if either the sequence $\{\epsilon_N\}$ has property (3), or there exists a sequence $\{y_n\}$ with property (3) such that $|\epsilon_n| < |y_n|$ for all $n$. For the two methods of interest the following is known.

---

[1] A more inclusive definition of order of convergence, referred to as the Q-order of convergence, might be

$$p = \sup\left\{ q : \limsup_{n\to\infty} \frac{|\delta_N|}{|\delta_{N-1}|^q} = 0 \right\} .$$

[2] This quantity should be evaluated on the assumption that numbers are represented to infinite precision, without rounding error.

 

(i) *Binary search* is first-order convergent.

(ii) *Fixed-point iteration*, when convergent, is *in general* first-order convergent for a simple root, i.e. one with $F'(x_*) \neq 0$. However, Newton-Raphson iteration, when convergent, is second-order convergent for a simple root, but only first-order convergent for a multiple root

## Examples

The cases to be studied as examples are

$$F(x) \equiv 2x - 3\sin x + 5 = 0\,, \tag{4}$$

and

$$F(x) \equiv x^3 - 8.5x^2 + 20x - 8 = 0\,. \tag{5a}$$

Note that equation (5a) can be factorised and rewritten as

$$F(x) \equiv \left(x - \tfrac{1}{2}\right)(x-4)^2 = 0\,. \tag{5b}$$

**Question 1** Show, with the help of a graph, that equation (4) has exactly one root (which is in fact $-2.88323687\ldots$).

## Binary Search

**Programming Task:** write a program to solve equation (4) by binary search.Provide for termination of the iteration as soon as the truncation error is guaranteed to be less than $0.5 \times 10^{-5}$, and print out the number of iterations, $N$, as well as the estimate of the root. Run the program for a number of suitable starting values to check that it is working; include some of these results in your report.

**Question 2** Suppose that the rounding error in evaluating $F(x)$ in equation (4) is at most $\delta$ for $|x| < \pi$. By considering a Taylor expansion of $F(x)$ near $x_*$, or otherwise, estimate the accuracy that may be expected for the calculated value of the root.

*Hint:* note that $|F'(x)| > 4$ for $-5\pi/4 < x < -3\pi/4$.

## Fixed-Point Iteration

There are many possible choices of $f$, e.g.

$$f(x) = x - h(F(x))\,, \tag{6}$$

for some function[3] $h(F)$ such that $h(0) = 0$.

**Programming Task:** write a program to implement the iteration scheme in equation (1) for general $f$. Provide for termination of the process as soon as $|x_N - x_{N-1}| < \epsilon$ or when $N = N_{max}$, whichever occurs first. Print out the values of $N$ and $x_N$ for each $N$, so that you can watch the progress of the iteration.

[3] Or *functional*.

 

**Question 3** Use the program to solve (4) by fixed-point iteration by taking

$$h(F) = \frac{F}{2+k} \tag{7a}$$

in (6), so that

$$f(x) = \frac{3\sin x + kx - 5}{2+k}\,, \tag{7b}$$

for some constant $k$.

(i) First run the program with $k = 0$, $\epsilon = 10^{-5}$, $x_0 = -2$, $N_{max} = 10$. Plot $y = f(x)$ and $y = x$ on the same graph, and use these plots to show why convergence should not occur. Explain the divergence by identifying a theoretical criterion that has been violated.[4]

(ii) Determine the values of $k$ for which convergence is guaranteed if $x_N$ remains in the range $(-\pi, -\pi/2)$.

(iii) Choose, giving reasons, a value of $k$ for which *monotonic* convergence should occur near the root, and also a value for which *oscillatory* convergence should occur near the root. Verify that these two values of $k$ give the expected behaviour, by running the program with $N_{max} = 20$.

(iv) Also run the case $k = 16$. This should converge only slowly, so set $N_{max} = 50$. Discuss whether the truncation error is expected to be less than $10^{-5}$ in this case?

(v) Discuss whether your results are consistent with first-order convergence.

**Question 4** Now use your program to find the *double root* of equation (5a) by fixed-point iteration by taking

$$h(F) = \tfrac{1}{20}\, F\,, \tag{8a}$$

in (6), so that

$$f(x) = \tfrac{1}{20}(-x^3 + 8.5x^2 + 8)\,. \tag{8b}$$

By considering $f'(x_*)$ explain why convergence will be slow at a multiple root for any choice of differentiable function $h$ in (6).

In your calculations some care may be needed over the choice of $x_0$. Also,

(a) since convergence will be slow, take $N_{max} = 1000$;

(b) suppress the printing of each iterate, but print out the *final* values of $N$ and $x_N$.

Is this an example of first-order convergence? Does the termination criterion ensure a truncation error of less than $10^{-5}$?

*Note:* it can be shown that the truncation error $\epsilon_N$ is asymptotic to $40/(7N)$ as $N \to \infty$.

## Newton-Raphson Iteration

A refinement of (6) is to let $h$ depend on the derivatives of $F$, i.e.

$$f(x) = x - h(F, F', F'', \ldots)\,. \tag{9a}$$

In Newton-Raphson iteration

$$h = \frac{F}{F'}\,. \tag{9b}$$

[4] The references at the end may prove helpful.

 

**Programming Task:** modify your program to recalculate the root of equation (4), and the double root of equation (5a), using Newton-Raphson iteration.

**Question 5** For equation (4), experiment with various $x_0$ until you have demonstrated a case that converges, and also a case that has not converged in 10 iterations. In the unconverged case, show graphically what happened in the first few iterations.

For both equation (4) and equation (5a) do your (converged) results bear out the theoretical orders of convergence? Comment on the effects of rounding error.

*Hint:* you may want to use a smaller value for $\epsilon$.

## References


[1] Epperson, J.F., *An Introduction to Numerical Methods and Analysis*, John Wiley & Sons (2007). ISBN-13: 9780470049631.

[2] Kharab, A. and Guenther, R.B., *An Introduction to Numerical Methods: A MATLAB Approach*, Second Edition, CRC Press (2005). ISBN-13: 9781584885573

[3] Press, W.H., Teukolsky, S.A. and Vetterling, W.T. and Flannery, B.P., *Numerical Recipes: The Art of Scientific Computing*, Third Edition, Cambridge University Press (2007). ISBN-10: 0521880688.

[4] Süli, E. and Mayers, D., *An Introduction to Numerical Analysis*, Cambridge University Press (2003). ISBN-10: 0521810264 (hardback), ISBN-10: 0521007941 (paperback).


 

# 1.1 Random Binary Expansions

*This project requires an understanding of the Part IA Probability and Part IA Analysis courses.*

Let $U = (U_1, U_2, \dots)$ represent an infinite sequence of coin tosses, with $U_i = 1$ if the $i$th toss is heads and $U_i = 0$ if it is tails. Suppose the coin tosses are independent, and that the probability of heads is $p \in (0, 1)$ and the probability of tails is $q = 1 - p$.

Given such a sequence we can define a real-valued random variable $X = f(U)$, taking values in the interval $[0, 1]$, by

$$f(U) = \sum_{i=1}^{\infty} \frac{U_i}{2^i}.$$

We may think of $U$ as a binary expansion of $X$ (though in fact some $x \in [0, 1]$ do not have a unique binary expansion, to wit, $0.1 = 0.01111\dots$).

Define the cumulative distribution function

$$F(x) = \mathbb{P}(X \leqslant x).$$

For most values of $p$ the function $F$ is pathological, but it does have some interesting properties.

## Approximating $F$

One way to approximate $F$ is by Monte Carlo simulation, as follows. Fix $n \in \mathbb{N}$. Generate a finite sequence $U^n = (U_1, \dots, U_n)$, and compute $X^n = \sum_{i \leqslant n} U_i/2^i$. Repeat this $N$ times to generate a random sample $X_1^n, \dots, X_N^n$. Now we can plot the *empirical* cumulative distribution function

$$\hat{F}(x) = \frac{1}{N} \sum_{j=1}^{N} 1[X_j^n \leqslant x]$$

where $1[A]$ is the indicator function for the event $A$. This should approximate the actual cumulative distribution function $F(x)$.

> **Question 1** Write a program to generate such a random sample. Plot the empirical distribution function for $p = 2/3$, using $n = 30$ and $N$ suitably large.

## Calculating $F$

It turns out that for some values of $x$, we can calculate $F(x)$ explicitly.

> **Question 2** Suppose that
>
> $$x = \sum_{i=1}^{n} \frac{x_i}{2^i}$$
>
> for some $n \in \mathbb{N}$ and some sequence $x_1, \dots, x_n$. (When this is so, we say $x$ has a finite binary expansion.) Find a formula for $F(x)$.

> **Question 3** Use your formula to plot a graph of $F$, for $p = 3/4$ and $n = 11$, sampling $F(x)$ at $x = 0, 1/2^n, 2/2^n, 3/2^n, 4/2^n, \dots, 1$. Comment briefly on how this graph compares to the graph you obtained in Question 1. Include also a short comparison of the complexity (number of time steps needed) of both algorithms for general $n$ and $N$.

 

## Properties of $F$

The plots you have produced should make you wonder: is $F$ continuous? Is it differentiable?

> **Question 4** Let $c$ have a finite binary expansion. Prove that $F(x)$ is continuous at $x = c$. Do your plots suggest that $F$ is continuous elsewhere? Prove or disprove.

We say that $F$ is left-differentiable at $c$ if the limit

$$\lim_{\delta \uparrow 0} \frac{F(c+\delta) - F(c)}{\delta}$$

exists and is finite, and that it is right-differentiable if the limit

$$\lim_{\delta \downarrow 0} \frac{F(c+\delta) - F(c)}{\delta}$$

exists and is finite. If $F$ is both left-differentiable and right-differentiable at $c$ and the two limits are equal, we say $F$ is differentiable at $c$.

> **Question 5** Let $p = 3/4$ and $c = 9/16$. Plot $(F(c+\delta) - F(c))/\delta$ against $\delta$ for a suitable range of values of $\delta$ for which $c + \delta$ has a finite binary expansion. Does your plot suggest that $F$ is left-differentiable or right-differentiable at $c$?

> **Question 6** Make a conjecture about whether $F$ is left-differentiable and/or right-differentiable at an arbitrary point $c$ with a finite binary expansion, for arbitrary $p \in (0, 1)$. Generate two or three plots which support your conjecture. Prove your conjecture.

 

**Project 1.1: Random Binary Expansions**

**Marking Scheme and additional comments for the Project Report**

*The purpose of these additional comments is to provide guidance on the structure and length of your CATAM report. Use the same concepts to write the rest of the reports. To help you assess where marks have been lost, this marking scheme will be completed and returned to you during Lent Term. You are advised to keep a copy of your write-up in order to correlate your answers to the marks awarded.*

| **Question no.** | **marks available**[1] | **marks awarded**[2] |
|---|---|---|
| **Programming task** *Program:* for instructions regarding printouts and what needs to be in the write-up, refer to the introduction to the manual. | | |
| **Question 1** *Comments:* Explain why you are satisfied that your choice of $N$ is reasonable [*approx. 5 lines* ][3] | C2.5 | |
| **Question 2** *Comments:* Do not include trivial steps in your answer [approx 10 lines] [3] | M1 | |
| **Question 3** *Comments:* Plot the graph obtained from the exact values of $F$, and use a separate graph (or two) for the comparison to the empirical distribution. In your comments try to argue quantitatively if possible. [*approx. 5 lines* ][3]. Argue briefly for the complexity [*approx. 3 lines* ][3] | C1.5+ M1 | |
| **Question 4** *Comments:* Do not include trivial steps in your answer [*approx.1 page* ][3] | M3 | |
| **Question 5** *Comments:* Scale the axes appropriately, and include a brief justification of your choice. [*approx. 2 lines* ][3] | C2 | |
| **Question 6** *Comments:* Do not include trivial steps in your proof. [*approx. 1 page* ][3] | C2+M5 | |
| **Excellence marks**. These are awarded for, *among other things*, mathematical clarity and good, clear output (graphs and tables) — see the introduction to the Project Manual. | E2 | |
| Total Raw Marks | 20 | |
| **Total Tripos Marks** | **40** | |

[1] C#, M# and E#: *Computational*, *Mathematical* and *Excellence* marks respectively.
[2] For use by the assessor
[3] This figure is only meant to be indicative of the length of your answer, rather than the exact number of lines you are expected to write

 

# 1.2 Ordinary Differential Equations

*This project builds on material covered in the Part IA Differential Equations and the Part IA lectures on Computational Projects.*

## 1 Numerical solution of first order ODEs

This section describes two methods for step-by-step integration of ordinary differential equations (ODEs) of the form

$$\frac{dy}{dx} = f(x, y)\,. \tag{1a}$$

An initial condition is specified at, say, $x = x_0$, i.e.

$$y\,(x_0) = Y_0\,. \tag{1b}$$

The numerical methods to be investigated are as follows:

(a) The **Euler** (or more precisely **forward Euler**) method employs the scheme

$$Y_{n+1} = Y_n + hf(x_n, Y_n)\,, \tag{2}$$

where $Y_n$ denotes the numerical solution at $x_n \equiv x_0 + nh$, that is, the solution at the $n$th step with step length $h$, starting from $x_0$ and $Y_0$.

*Definition.* The *global error* after the $n$th step is defined as

$$E_n = Y_n - y(x_n)\,, \tag{3a}$$

where $y(x_n)$ is the exact solution to (1a) at $x_n$.

*Definition.* The *local error* of the first step is defined as $e_1 = Y_1 - y(x_1)$. For subsequent steps the *local error* is defined as

$$e_n = Y_n - w(x_n)\,, \tag{3b}$$

where $w(x_n)$ is the exact solution to (1a) at $x = x_n$ starting from $x = x_{n-1}$ and $y = Y_{n-1}$. Note that, in general, $Y_{n-1} \neq y(x_{n-1})$ for $n > 1$.

For the Euler method it can be shown that $e_n$ is $O(h^2)$ as $h \to 0$. As a result the Euler method is said to have *first-order accuracy.*

(b) The fourth-order **Runge–Kutta** (RK4) method employs the scheme:

$$Y_{n+1} = Y_n + \tfrac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)\,, \tag{4a}$$

where

$$k_1 = hf(x_n,\ Y_n)\,, \tag{4b}$$
$$k_2 = hf(x_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{2}k_1)\,, \tag{4c}$$
$$k_3 = hf(x_n + \tfrac{1}{2}h,\ Y_n + \tfrac{1}{2}k_2)\,, \tag{4d}$$
$$k_4 = hf(x_n + h,\ Y_n + k_3)\,. \tag{4e}$$

The RK4 method has *fourth-order accuracy*, i.e. $e_n$ is $O(h^5)$ as $h \to 0$.

Theoretical background for the stability and accuracy of these methods can be found in, for example, *An Introduction to Numerical Methods and Analysis* by J.F.Epperson, *An Introduction to Numerical Methods* by A.Kharab and R.B.Guenther and *Numerical Recipes* by Press *et al.*

 

## 2 Stability and accuracy of the numerical methods

This section will explore numerical integration of equation (1a) with

$$f(x,y) = -4y + 3\mathrm{e}^{-x} \tag{5a}$$

and initial condition

$$y(0) = 0\ . \tag{5b}$$

This has the exact solution

$$y(x) = \mathrm{e}^{-x} - \mathrm{e}^{-4x}\ . \tag{6}$$

**Programming Task:** Write program(s) to apply the Euler and RK4 methods to this problem.

### 2.1 Stability

**Question 1** Using the Euler method, starting with $Y_0 = 0$, compute $Y_n$ for $x$ up to $x = 9$ with $h = 0.6$, i.e. for $n$ up to $10/h = 15$. Tabulate the values of $x_n$, the numerical solution $Y_n$, the analytic solution $y(x_n)$ from (6), and the global error $E_n \equiv Y_n - y(x_n)$. You should find that the numerical result is unstable: the error oscillates with a magnitude that ultimately grows proportional to $\mathrm{e}^{\gamma x}$, where the 'growth rate' $\gamma$ is a positive constant which you should estimate.

Examine the stability of the method by repeating the calculation for several values of $h$, presenting only a judicious selection of output to illustrate the behaviour. What effect does reducing $h$ have on the size of the instability, and on its growth rate?

**Question 2**
(i) Find the analytic solution of the Euler *difference* equation

$$Y_{n+1} = Y_n + h\left(-4Y_n + 3\left(\mathrm{e}^{-h}\right)^n\right) \quad \text{with} \quad Y_0 = 0\,. \tag{7}$$

(ii) Hence explain why and when instability occurs, and with what growth rate.

(iii) Show that in the limit $h \to 0$, $n \to \infty$ with $x_n \equiv nh$ fixed, the solution of the difference equation (7) converges to the solution (6) of the differential equation specified by (1a), (5a) and (5b).

### 2.2 Accuracy

**Question 3** Integrate the ODE specified by specified by (1a), (5a) and (5b) numerically with $h = 0.2$ from $x = 0$ to $x = 4$ using both the Euler and RK4 methods. Plot $Y_n$ against $x_n$ for each method with the exact solution (6) superposed.

**Question 4** For both the Euler and the RK4 methods, tabulate the global error $E_n$ at $x_n = 0.4$ against $h \equiv 0.4/n$ for $n = 2^k$ with $k = 0, 1, 2, \ldots, 15$, and plot a log–log graph of $|E_n|$ against $h$ over this range.

Comment on the relationship of your results to the theoretical accuracy of the methods.

 

## 3 Numerical solutions of second-order ODEs

The same time-stepping methods can also be applied to higher order ODEs. This section will explore solutions to a damped harmonic oscillator with a driving force. Specifically, start by considering the following equation

$$\frac{d^2y}{dt^2} + \gamma\frac{dy}{dt} + y = \sin(\omega t), \tag{8}$$

where $\gamma$ and $\omega$ are non-negative real constants and $t$ and $y$ are real variables.

> **Question 5** Find the general solution of (8) for the lightly damped case, $0 < \gamma < 2$. Show that
>
> $$y \to A_s \sin(\omega t - \phi_s) \quad \text{as } t \to \infty \tag{9}$$
>
> where the 'steady-state' amplitude $A_s$ and 'steady-state' phase shift $\phi_s$ are to be found in terms of $\gamma$ and $\omega$.

Equation (8) can be rewritten as a pair of coupled first-order ODEs for

$$y^{(1)}(t) \equiv y(t) \qquad \text{and} \qquad y^{(2)}(t) \equiv \frac{dy(t)}{dt} \ , \tag{10}$$

namely

$$\frac{dy^{(1)}}{dt} = f^{(1)}(t, y^{(1)}, y^{(2)}) \equiv y^{(2)} \ , \tag{11}$$

$$\frac{dy^{(2)}}{dt} = f^{(2)}(t, y^{(1)}, y^{(2)}) \equiv -\gamma y^{(2)} - y^{(1)} + \sin(\omega t). \tag{12}$$

This system of equations can be solved using either the Euler or the RK4 method, but here we will just use the latter.

The RK4 method can be generalised to solve first order systems of equations by writing

$$\mathbf{Y}_{n+1} = \mathbf{Y}_n + \tfrac{1}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)\,, \tag{13a}$$

where

$$\mathbf{k}_1 = h\mathbf{f}(x_n,\ \mathbf{Y}_n)\,, \tag{13b}$$

$$\mathbf{k}_2 = h\mathbf{f}(x_n + \tfrac{1}{2}h,\ \mathbf{Y}_n + \tfrac{1}{2}\mathbf{k}_1)\,, \tag{13c}$$

$$\mathbf{k}_3 = h\mathbf{f}(x_n + \tfrac{1}{2}h,\ \mathbf{Y}_n + \tfrac{1}{2}\mathbf{k}_2)\,, \tag{13d}$$

$$\mathbf{k}_4 = h\mathbf{f}(x_n + h,\ \mathbf{Y}_n + \mathbf{k}_3) \tag{13e}$$

where $\mathbf{f}$, $\mathbf{Y}$, and $\mathbf{k}$ are two-dimensional vectors and the components of the vectors $\mathbf{f}$ and $\mathbf{Y}$ can be written $\mathbf{f} = (f^{(1)}, f^{(2)})$ and $\mathbf{Y} = (Y^{(1)}, Y^{(2)})$.

> **Programming Task:** Write a program to solve equation (8) using the RK4 method, subject to the initial conditions
>
> $$y = \frac{dy}{dt} = 0 \quad \text{at } t = 0\ . \tag{14}$$

 

**Question 6** Taking $\gamma = 1$ and $\omega = \sqrt{3}$, use your program to compute $Y_n$ for $t$ up to 10 with $h = 0.4$ [i.e. for $n$ up to 25], and tabulate the numerical solution $Y_n$, the analytic solution $y(t_n)$ and the global error $E_n \equiv Y_n - y(t_n)$ against $t_n$. Repeat with both $h = 0.2$ and $h = 0.1$ [integrating up to $t = 10$, i.e. for $n$ up to 50 and 100 respectively], not necessarily presenting all the output. Comment on the errors.

**Question 7** Use your RK4 program (with suitable $h$) to generate and plot numerical solutions of (8) and (14) up to $t = 40$ for $\omega = 1$ and $\gamma = 0.25$, 0.5, 1.0 and 1.9, checking that they agree with the analytic solutions. Do likewise for $\omega = 2$ and the same values of $\gamma$. Explain the differences between the various cases in terms of the mathematics and the physics of the system under investigation.

The last question considers a case with nonlinear damping,

$$\frac{d^2y}{dt^2} + \frac{d}{dt}\left(\tfrac{1}{3}\delta^3\, y^3\right) + y = \sin t \ , \tag{15}$$

for which an analytic solution is not available. The initial conditions are as before,

$$y = \frac{dy}{dt} = 0 \quad \text{at } t = 0 \ . \tag{16}$$

**Question 8** For $\delta = 0.25$, 0.5, 1.0 and 20, use your RK4 program to generate and plot numerical solutions to (15)–(16) for $t$ up to 60, using suitable value(s) of $h$ (justify your choice). Comment on the solutions, comparing them with each other and with those of Question 7 for $\omega = 1$.

*Hint*: it *may* be helpful to observe that when $\delta$ is 'small', equation (15) has a $2\pi$-periodic solution of the form

$$y = \sum_{n=-1}^{\infty} \delta^n y_n(t) \tag{17}$$

where each $y_n(t)$ is periodic in $t$ with period $2\pi$ and

$$y_{-1}(t) = A\cos t \ , \quad y_0(t) = B\sin t + C\sin 3t \tag{18}$$

for suitable values of the constants $A$, $B$ and $C$ [recall that $\cos^3\theta = \frac{3}{4}\cos\theta + \frac{1}{4}\cos 3\theta$, and note that to determine $y_0$ completely it is necessary to consider terms of order $\delta$]. What if $\delta$ is 'large'?

## Additional Reference

Boyce, W. E., and DiPrima, R. C., 2001, *Elementary Differential Equations and Boundary Value Problems*, 7th edition. Publ. John Wiley & Sons Inc.

**Project 1.2: Ordinary Differential Equations**

**Marking Scheme and additional comments for the Project Report**

*The purpose of these additional comments is to provide guidance on the structure and length of your CATAM report. Use the same concepts to write the rest of the reports. To help you assess where marks have been lost, this marking scheme will be completed and returned to you during Lent Term. You are advised to keep a copy of your write-up in order to correlate your answers to the marks awarded.*

| **Question no.** | **Marks available**[1] | **Marks awarded**[2] |
|---|---|---|
| **Programming task** *Program:* for instructions regarding printouts and what needs to be in the write-up, refer to the introduction to the manual. | | |
| **Question 1** *Tables:* for presentation and layout, refer to the introduction. [*quarter page*][3] | C2 | |
| **Question 2** *Analytic solution:* do not include trivial steps in your working. [*approx. 15 lines*][3] | M2 | |
| **Question 3** *Graphs:* you may use one graph or two. | C1 | |
| **Question 4** *Graphs:* ditto.<br>*Comments:* what can be said about how the global error $E_n$ for each method varies with $h$? How is this reflected in the plots? [*approx. 3 lines*][3] | C1<br>M1 | |
| **Question 5** *Analytic solution:* do not include trivial steps in your working; be sure to state the results unambiguously. [*approx. 5 lines*][3] | M1 | |
| **Question 6** *Analytic and numerical solutions compared:* the purpose of this step is to check that the program works and gives accurate answers ('validation'). Do the errors behave as expected when $h$ is decreased? [*quarter page*][3] | C2+M2 | |
| **Question 7** *Comments:* first identify the salient features of the plots. Examine the nature of the functions that you are plotting: what are their components and how do these contribute to the overall solutions? Then use mathematical arguments (cf. the Part IA course *Differential Equations*) to explain the behaviour of the plots; link to the theory of the physical system under investigation. [*one page*][3] | C1+M2 | |
| **Question 8** *Numerical solutions:* explain why you are satisfied that your chosen value(s) of $h$ will deliver sufficiently accurate results.<br>*Comments:* identify the key similarities and differences between the various solutions, and with the help of the hint, or otherwise, try to explain them mathematically and/or physically. [*one page*][3] | C1<br><br>M2 | |
| **Excellence marks** awarded for, *among other things*, mathematical clarity and good, clear output (graphs and tables) — see the introduction to the project manual. | E2 | |
| **Total Raw Marks** | 20 | |
| **Total Tripos Marks** | **40** | |

[1] C#, M# and E#: *Computational*, *Mathematical* and *Excellence* marks respectively.
[2] For use by the assessor.
[3] Your aim is to answer succinctly the questions including graphs and tables, and to make all important points. The length specified here should b sufficient for you to do this but is not a target.

 

# 2.1 The Restricted Three-Body Problem

*Knowledge of the Part IA Dynamics and Relativity lecture course is useful for this project.*

## 1 Introduction

Determining the motion of gravitating bodies is a classical problem. For two bodies there is a stable analytic solution which describes rotation about the joint centre of mass. For three bodies, the problem cannot be solved analytically. Various simplifications have historically been considered, one of which is the 'restricted three-body problem' in which the third body is taken to be much smaller in mass than the other two and therefore does not affect their motion. The problem is then to solve for the motion of the third body, under the gravitational influence of the other two.

It is convenient to transform to a rotating frame of reference in which the first two bodies appear stationary and the origin corresponds to their joint centre of mass. Scalings may be chosen so that the angular velocity of this frame is 1 and the distance between the two bodies is 1. The only parameter then appearing is the quantity $\mu$ defined such that the two masses are in the ratio $\mu : 1-\mu$ and are situated respectively at the points $(\mu-1, 0)$ and $(\mu, 0)$. These points are denoted by $P_1$ and $P_2$. At time $t$, the third body has position $(x(t), y(t))$. Its equation of motion is

$$\ddot{x} - 2\dot{y} = -\frac{\partial \Omega}{\partial x}\,, \tag{1}$$

$$\ddot{y} + 2\dot{x} = -\frac{\partial \Omega}{\partial y}\,, \tag{2}$$

where

$$\Omega = -\tfrac{1}{2}\mu r_1^2 - \tfrac{1}{2}(1-\mu) r_2^2 - \frac{\mu}{r_1} - \frac{1-\mu}{r_2}\,, \tag{3}$$

with $r_1^2 = (x+1-\mu)^2 + y^2$ and $r_2^2 = (x-\mu)^2 + y^2$.

Despite the substantial restriction to the full three-body problem which this represents, it is not possible to solve (1) and (2) analytically.

> **Question 1** Show from (1) and (2) that the quantity
>
> $$J = \tfrac{1}{2}\dot{x}^2 + \tfrac{1}{2}\dot{y}^2 + \Omega(x, y)$$
>
> is constant following the motion. Deduce that trajectories must be confined to the region
>
> $$\Omega(x, y) \leqslant \Omega(x_0, y_0) + \tfrac{1}{2}u_0^2 + \tfrac{1}{2}v_0^2\,, \tag{4}$$
>
> where $x_0$, $y_0$, $u_0$ and $v_0$ are the initial values of $x$, $y$, $\dot{x}$ and $\dot{y}$, respectively.

> **Programming Task:** Write a program to solve (1) and (2) numerically, given suitable initial conditions on $x$, $y$, $\dot{x}$ and $\dot{y}$. You may wish to use a suitable MATLAB ODE solver, such as `ode45`, which automatically adjust the time-step. You may need to adjust the error tolerance of your solver. If you use a different programming language and a fixed-step integration routine, you may need to use very small time-steps.
>
> Write a second program to plot contours of $\Omega(x, y)$ in the $(x, y)$ plane.

 

Throughout this project, it will be necessary to check that your program is yielding accurate results. Standard checks include (i) testing the program against known analytic solutions (if there are any), and (ii) varying the time-step or error tolerance. The fact that $J$ is constant provides another possible check on the accuracy of the numerical solution. In your report, comment on the checks you have performed.

## 2 Space travel

Assume that the third body is a spacecraft, with the first two bodies being planets of equal mass ($\mu = 0.5$).

**Question 2** Consider motion in the neighbourhood of $P_2$, so that the effect of $P_1$ may be ignored and (3) may be approximated by

$$\Omega = -\frac{1}{2r_2} \, . \tag{5}$$

Show that (1) and (2) then have analytic solutions with the spacecraft in a circular orbit of radius $a$ about $P_2$, where $a$ can take any value. [The approximation (5) is of course only valid if $a$ is small.]

Modify your program to solve (1) and (2) with $\Omega$ specified by (5) instead of (3). Demonstrate (for one value of $a$) that the modified program can reproduce the analytic solutions.

**Question 3** Now return to the original problem (1)–(3) with $\mu = 0.5$, and take initial conditions $x = 0.2$, $y = 0$, $\dot{x} = 0$, $\dot{y} = v_0$ with $v_0 = -0.50$, $-1.00$, $-1.04$, $-1.18$, and $-1.25$, $-1.50$ in turn. Integrate from $t = 0$ to $t = 15$. For each case, make a plot that shows the trajectory, together with the allowed region $\Omega(x, y) \leqslant J$ (or its complement). Give an indication of the accuracy of your numerical integrations.

Comment on the trajectories, and how these and the allowed region change as $v_0$ increases. Is the allowed region a useful guide to the size of the trajectory? How does a trajectory behave close to the boundary of the allowed region? Which value of $v_0$ would be most suitable to travel from the neighbourhood of $P_2$ to the neighbourhood of $P_1$?

## 3 Lagrange points and asteroids

*In this part of the project do not restrict attention to the case $\mu = 0.5$.*

**Question 4** By examining contour plots of $\Omega$ show that the system (1)–(3) generally has five equilibrium points. (Three lie on the $x$-axis and the other two — the *equilateral Lagrange points* — at the third vertex of an equilateral triangle whose other two vertices are at $P_1$ and $P_2$.) Display contour plots for three values of $\mu$, with the equilibrium points marked.

Investigate the stability of the different equilibrium points numerically, i.e., by starting trajectories a small distance away from the equilibrium point and integrating forward in time. Display plots of some representative trajectories in the $(x, y)$ plane to illustrate your results. (You may also find it useful to show plots of $x$ or $y$ against $t$.)

 

What do you conclude about the stability of the equilibrium points on the $x$-axis? Do the stability properties depend on $\mu$? Confirm your findings by performing a linearised stability analysis about such points. You should be able to deduce any necessary information about the second derivatives of $\Omega$ by considering the shapes of the contours, rather than by detailed calculation.

**Question 5** Continue with a numerical investigation of the stability of the equilateral Lagrange points, for parameter values $\mu = 0.008$, $0.022$, $0.044$, $0.08$ and $0.5$. Illustrate the results in your write-up with one trajectory picture for each.

How do the stability properties change with $\mu$? Either by numerical experimentation, or by performing a linearised stability analysis, find (to two significant figures) the critical value $\mu_c$ dividing values of $\mu$ for which the point is stable from those for which it is unstable. [This time the linearised stability analysis does require calculation of the second derivatives of $\Omega$.] For the stable cases give a qualitative description of the motion.

**Question 6** The Trojans are a group of asteroids observed at the Sun–Jupiter equilateral Lagrange point. For the Sun-Jupiter system $\mu = 9.54 \times 10^{-4}$. Is the persistence of the Trojans at this point consistent with your findings above? The Earth–Moon system has $\mu = 0.012141$. Would you expect analogues of the Trojans at this point?

 

# 2.2 Parallel-Plate Capacitor: Laplace's Equation

*This project is self-contained. Part IB Electromagnetism provides background but is not necessary.*

## 1 Introduction

You will solve Laplace's equation in two dimensions, in order to compute properties of a simple capacitor. The system is sketched in Fig. 1, it consists of two parallel rectangular plates of size $\ell_x \times \ell_z$, separated by a distance $d$. The thickness of the plates is negligible. Let the electric potential at position $(x, y, z)$ be $\varphi(x, y, z)$. The centres of the plates are at $(x, y, z) = (0, \pm\frac{d}{2}, 0)$. On the upper plate then $\varphi = +V/2$ and on the lower plate then $\varphi = -V/2$.

The problem is to find a function $\phi(x, y)$ that solves

$$\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0 \tag{1}$$

subject to boundary conditions:

$$\begin{aligned} &\phi = (\pm V/2) \text{ on the plates: } y = (\pm d/2) \text{ and } |x| < (\ell_x/2) \\ &\phi \to 0 \text{ as either } |x| \to \infty \text{ or } |y| \to \infty \end{aligned} \tag{2}$$

Assume that $\ell_z \gg \ell_x$: then the solution $\phi(x, y)$ is an accurate approximation to $\varphi(x, y, 0)$, which is the potential in the plane $z = 0$.

The electric field (in this plane) is given by

$$\mathbf{E} = -\nabla\phi = \left(-\frac{\partial\phi}{\partial x}, -\frac{\partial\phi}{\partial y}\right) . \tag{3}$$

It is useful to work in dimensionless units, defined as follows:

$$X = \frac{2x}{d} , \qquad Y = \frac{2y}{d} , \qquad L = \frac{\ell_x}{d} , \qquad \Phi = \frac{\phi}{V} , \tag{4}$$

where $q$ is computed for the upper plate. This simplifies our problem to

$$\frac{\partial^2\Phi}{\partial X^2} + \frac{\partial^2\Phi}{\partial Y^2} = 0 , \tag{5}$$

with $\Phi = \pm\frac{1}{2}$ on the plates, which are at $Y = \pm 1$ with $|X| \leqslant L$; also $\Phi \to 0$ far from the plates.

For large $L$, the expected physical behaviour *in the gap between the plates* is that $\nabla\Phi$ is (almost) aligned with the $Y$-direction, and depends weakly on $X, Y$.

## 2 Numerical Method

For numerical purposes, it is convenient to solve Eq. (5) in a rectangular domain $\mathcal{D} = [-D_X, D_X] \times [-D_Y, D_Y]$ where $D_X, D_Y$ are parameters to be chosen appropriately. We set $\Phi = 0$ on the boundary of this domain*: the solution to the original problem is recovered by taking $D_X, D_Y \to$

*This corresponds physically to putting the capacitor in a conducting box at potential $\phi = 0$.

 

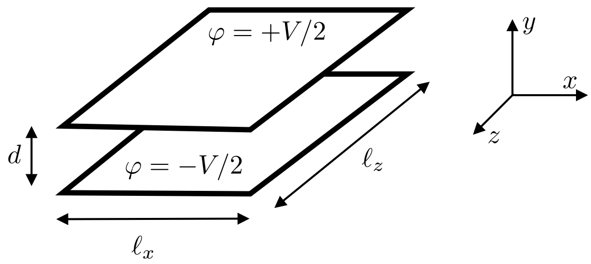


Figure 1: Sketch of two parallel plates of size $\ell_x \times \ell_z$, separated by distance $d$. The upper plate is at potential $\varphi = V/2$ and the lower one is at $\varphi = -V/2$.

$\infty$. The solution $\Phi$ has symmetries $\Phi(-X, Y) = \Phi(X, Y)$ and $\Phi(X, -Y) = -\Phi(X, Y)$. Hence it is sufficient to compute $\Phi$ in the positive quadrant of $\mathcal{D}$.

Within the domain $\mathcal{D}$, define a grid of points $(X_m, Y_n) = (mh, nh)$ where $m, n$ are integers and $h$ is the grid spacing. The domain $\mathcal{D}$ should be chosen such that $N_x = D_X/h$ and $N_y = D_Y/h$ are integers, as is $1/h$. You will compute a numerical approximation to $\Phi(X_m, Y_n)$ which is denoted by $\Phi_{m,n}$.

> **Question 1** Suppose that we can find a numerical solution [for all grid points $(X_i, Y_j)$] of the equation
>
> $$\Phi_{i-1,j} + \Phi_{i+1,j} + \Phi_{i,j-1} + \Phi_{i,j+1} - 4\Phi_{i,j} \;=\; 0 \tag{6}$$
>
> subject to appropriate boundary conditions. Show that as $h \to 0$, this numerical solution approximates a solution of the Laplace equation (5).

Eq. 6 can be solved by iteration. Starting with an initial guess $\Phi^{(0)}$ one defines a sequence of approximations $\Phi^{(k)}$ with $k = 1, 2, \ldots$, which converge to the solutions of Eq. (6). For some mesh points, the value of $\Phi_{m,n}$ is fixed by boundary conditions (either from the plates or from the boundary of $\mathcal{D}$). For the remaining points, an iteration rule is required.

A simple rule is the Jacobi scheme

$$\Phi^{(k+1)}_{i,j} \;=\; \frac{1}{4}\left[\Phi^{(k)}_{i-1,j} + \Phi^{(k)}_{i+1,j} + \Phi^{(k)}_{i,j-1} + \Phi^{(k)}_{i,j+1}\right] \tag{7}$$

However, it is more efficient in practice to use the successive over-relaxation (SOR) method:

$$\Phi^{(k+1)}_{i,j} = (1-\omega)\Phi^{(k)}_{i,j} \;+\; \frac{\omega}{4}\left[\Phi^{(k+1)}_{i-1,j} + \Phi^{(k)}_{i+1,j} + \Phi^{(k+1)}_{i,j-1} + \Phi^{(k)}_{i,j+1}\right] \tag{8}$$

where $\omega$ is a parameter with $1 \leqslant \omega < 2$. The case $\omega = 1$ is called Gauss-Seidel iteration, larger $\omega$ corresponds to increasing "over-relaxation" which can be effective for accelerating convergence. Note that the right hand side of Eq. 8 mixes quantities from the $k$th and $(k+1)$th iterations, this is feasible in practice because the $\Phi^{(k+1)}_{m,n}$ are computed sequentially in $m, n$. Full details can be found in texts on numerical methods, such as Ref. [1].

## 3 Computing the potential and the electric field

When implementing the SOR method, you should restrict to the positive quadrant of $\mathcal{D}$, and you will need to take care with the iteration rule at the edge of this domain. You can use that $\Phi = 0$ on the boundaries $X = D_X$ and $Y = D_Y$. For $Y = 0$ then $\Phi = 0$ since $\Phi$ is odd in $Y$. For $X = 0$ then you should replace $\Phi^{(k+1)}_{-1,j}$ in the iteration rule by $\Phi^{(k)}_{1,j}$, using that $\Phi(-h, Y) = \Phi(h, Y)$ by symmetry.

Note carefully that the boundary conditions fix $\Phi$ for points on the plates, so the iteration rule should not be applied there.

You will also need a criterion for stopping the iteration. For this, define the residual

$$r_k = \frac{1}{N} \sum_i \sum_j \left| \Phi^{(k)}_{i,j} - \Phi^{(k-1)}_{i,j} \right| \tag{9}$$

where $N$ is the total number of mesh points, and the sum runs over all such points. The $k$th iteration $\Phi^{(k)}$ can be taken as a suitable approximation for the solution $\Phi$ if $r_k < \epsilon_{\text{tol}}$, where $\epsilon_{\text{tol}}$ is a small tolerance parameter that you will need to choose.

**Programming Task:** Write a program to implement the SOR iteration method. The program output will depend on parameters $L, D_X, D_Y, h, \omega, \epsilon_{\text{tol}}$. It will be necessary to plot the solution $\Phi$, either as a function of two variables, or as one-dimensional "slices" along the $x$ or $y$ directions. The validity of the method does not depend on your initial guess $\Phi^{(0)}$, you should verify this.

**Question 2** Test your program as follows. Take $L = 1$ and $h = \frac{1}{2}$ and $(D_X, D_Y) = (2, 2)$. Take $\omega = 1$. By suitably adjusting $\epsilon_{\text{tol}}$, verify that the solution to the discretised problem in Eq. (6) has $\Phi_{1,1} = \Phi_{1,3} = 0.238$ (to three significant figures). Show how your estimate of $\Phi$ depends on $(X, Y)$ inside $\mathcal{D}$.

The $Y$-component of the electric field (in dimensionless units) is $\mathcal{E}_Y = -\partial\Phi/\partial Y$, which can be estimated by a finite difference as

$$\mathcal{E}_Y(X, Y) \approx \pm\frac{1}{h}[\Phi(X, Y \mp h) - \Phi(X, Y)] \tag{10}$$

Note that $\mathcal{E}_Y$ may be discontinuous at $Y = 1$, in which case the left- and right-derivatives are not equal (but both can be estimated by choosing appropriately the $\pm$ signs).

**Question 3** Take $L = 2$ and $h = \frac{1}{4}$ and $(D_X, D_Y) = (4, 4)$. Plot the corresponding numerical approximations to $\Phi(0, Y)$ and $\Phi(2, Y)$ for $0 < Y < D_Y$. Plot estimates of the field on the mid-plane: $\mathcal{E}_Y(X, 0)$ for $0 < X < D_X$. Also plot the electric field on the upper and lower surfaces of the plate $\mathcal{E}_Y(X, Y \to 1)$. Comment on how you ensured that your results solve Equ. (6) to sufficient accuracy.

**Question 4** For the parameters of question 3, investigate the effect of reducing $h$ (for example, you might compare $h = \frac{1}{4}, \frac{1}{8}, \frac{1}{12}$). Plot the same quantities considered in that question, for several different values of $h$, on the same axes. Are your results consistent with convergence to a suitable solution of the Laplace equation, as $h \to 0$?

**Question 5** Still for the parameters of question 3, how many iterations of SOR are required for convergence? Investigate how this depends on $\omega$.

**Question 6** Recall that the original problem of interest was posed on the infinite domain $X, Y \in \mathbb{R}^2$ with boundary condition $\Phi \to 0$ as $|X|, |Y| \to \infty$. By varying $D_X$ and $D_Y$, investigate how much your numerical solutions near the plates are affected by the the boundary condition that $\Phi = 0$ on the boundary of $\mathcal{D}$.

 

## 4 Comparison with semi-infinite plates

To understand the behavior of $\Phi$ near the ends of the plates, it is useful to consider a slightly different problem, for which exact results are available. Instead of two finite plates, we consider semi-infinite ones corresponding to the line segments $Y = \pm 1$ with $X \in (-\infty, L]$. In this case, the theory of conformal mappings provides formulae for *equipotentials* (lines of constant $\Phi$).†

The theory defines a function $\Psi : \mathbb{R}^2 \to \mathbb{R}$ such that lines of constant $\Psi$ are everywhere orthogonal to the equipotentials.‡ Now define $W = -\Phi + \mathrm{i}\Psi$ (where $\mathrm{i} = \sqrt{-1}$, as usual). Then

$$(X - L) + \mathrm{i}Y = \frac{1 + \mathrm{e}^{-2\pi \mathrm{i} W}}{\pi} - 2\mathrm{i}W \ . \tag{11}$$

Equipotential lines can be computed (parametrically) from this formula by fixing some $\Phi \in [-\frac{1}{2}, \frac{1}{2}]$, varying $\Psi$, and taking real/imaginary parts. Lines of constant $\Psi$ (known as *field lines*) are obtained similarly, by fixing $\Psi$ and varying $\Phi$. Obviously, the value of $L$ only shifts these solutions along the $X$-direction.

In addition, the electric field $(\mathcal{E}_X, \mathcal{E}_Y) = (-\partial\Phi/\partial X, -\partial\Phi/\partial Y)$ can be obtained in terms of $\Phi, \Psi$ from the following formula:

$$-\frac{\partial \Phi}{\partial X} + \mathrm{i}\frac{\partial \Phi}{\partial Y} = \frac{\mathrm{i}}{2(\mathrm{e}^{-2\pi \mathrm{i} W} + 1)} \tag{12}$$

**Question 7** Consider the semi-infinite case with $L = 0$ and plot some illustrative equipotentials and field lines. (When plotting equipotentials, some care is required with the range of $\Psi$-values.) The upper surface of the top plate corresponds to $\Phi = \frac{1}{2}$ and $\Psi \in (0, \infty)$ while the lower surface is $\Phi = \frac{1}{2}$ and $\Psi \in (-\infty, 0)$: illustrate this by plotting some equipotentials with $\Phi = \frac{1}{2} - \delta$, for suitably small $\delta$.

**Question 8** Consider the electric field on the upper surface of the top plate, as follows: You know from Question 7 how $\Psi$ and $\Phi$ behave on the plate. Use Eq. (12) to show that the electric field on the surface is in the $Y$ direction and derive its magnitude in terms of $\Psi$. Consider the asymptotic behaviour of Eqs. (11,12) as $\Psi \to 0^+$ and $\Psi \to \infty$ (always with $\Phi = \frac{1}{2}$) and hence show that

$$\mathcal{E}_Y(X, 1^+) \approx \begin{cases} a(L - X)^{-1/2} & \text{as } X \to L^- \\ b(L - X)^{-1} & \text{as } X \to -\infty \end{cases}$$

where $a$ and $b$ are constants to be determined.

What happens on the lower surface of the plate?

**Question 9** For large values of $L, D_X, D_Y$, the numerical solutions for finite plates can be compared with the semi-infinite case. For a few values of $L$, make plots that compare your numerical estimates of the field on the plate surfaces with the results for semi-infinite plates derived in Question 8, which can again be plotted parametrically (by varying $\Psi$).

For the numerical solutions, you will need to choose values of $h, D_X, D_Y$ that balance the computational time with the accuracy required. In which parts of the domain do the numerical solutions depend most strongly on these parameters (and on $L$)? In which parts of the domain do the solutions match for finite and semi-infinite cases?

†A detailed discussion is given in Ref. [2], including some results for finite plates, but these are not needed for this project.

‡This $\Psi$ is a harmonic conjugate of $-\Phi$, which means that $\partial\Phi/\partial X = -\partial\Psi/\partial Y$ and $\partial\Phi/\partial Y = \partial\Psi/\partial X$.

 

## References


1. A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*, CUP, 1996.

2. H. B. Palmer, The Capacitance of a Parallel-Plate Capacitor by the Schwartz-Christoffel Transformation, *Transactions of the American Institute of Electrical Engineers*, vol.56(3), pp.363-366 (1937). See http://dx.doi.org/10.1109/T-AIEE.1937.5057547.


 

# 2.3 Curves in the Complex Plane

*This project uses material found in both the Complex Methods and Complex Analysis courses.*

## 1 Introduction

A *curve* $\gamma$ is a continuous map from the closed bounded interval $[0, 2\pi]$ (or $[0, 1]$ or even $[a, b]$) to $\mathbb{C}$. We say that $\gamma$ is *smooth* if $\gamma(t) = (u(t), v(t))$ for $u, v$ functions that are infinitely differentiable and $\gamma$ is *closed* if $\gamma(0) = \gamma(2\pi)$.

Suppose that $w = f(z)$ is a complex polynomial in $z$ and we take $r > 0$. If $C_r$ is the circle of radius $r$ then it is the case that $f(C_r)$ is a smooth closed curve in the $w$-plane. Curves that are generated in this way have a number of interesting properties that will be investigated in this project.

## 2 Complex roots of $f(z)$

In this section you will find the complex roots of the polynomial

$$f_1(z) = z^3 - z^2 + (2 - i)z - 1 - i \tag{1}$$

and of its first derivative $f_1'(z)$.

> **Programming Task:** Write a program that, given a polynomial $f(z)$, plots a graph of $f(C_r)$ and computes the coordinates and modulus of the closest point on $f(C_r)$ to $0 + 0i$. Your program should prompt you to enter a value for $r$.

> **Question 1** Using your program find the three roots of $f_1(z)$ to three significant figures and record the roots in your write-up, together with the corresponding values of $r$. There is no need for your program to automate the search for an $r$ such that $\min[|f(C_r)|] = 0$ — trial and improvement is an adequate method. Nevertheless, you may find it helpful to include an option to carry out the search automatically.
>
> Taking your output for the roots, how can you justify that these answers are indeed correct to three significant figures?

> **Question 2** Write down the first derivative $f_1'(z)$ of $f_1(z)$ and use your program to find the two roots of $f_1'(z)$ to three significant figures. Record the roots in your write-up, together with the corresponding values of $r$. Again, how is your answer justified?

## 3 Images $f(C_r)$ of $C_r$

In this section you will explore the geometry of the image $f(C_r)$ in the $w$-plane for some polynomials $f(z)$. Consider the polynomial $g(z) = z^3 + z$.

> **Question 3** Change the program that you wrote for Question 1 so that it plots the image $g(C_r)$ for a given $r$. Examine what happens as $r$ increases from a very small value to a moderately large one. In your write-up explain what happens. Use plots of $g(C_r)$ for suitably chosen values of $r$ to illustrate your explanation.

 

**Question 4** Repeat Question 3 for the polynomial $h(z) = z^3 - z^2 - z + 1$. Again use plots of $h(C_r)$, along with figures chosen to zoom in on relevant details of the image curves.

**Question 5** In the light of what you have found in Questions 3 and 4, explain what happens to the image curve $f_1(C_r)$ of the original polynomial $f_1$ in Question 1 as $r$ increases from a suitably small value to a large one. Again, use plots of $f_1(C_r)$ for suitably chosen values of $r$ to illustrate your explanation, including those chosen to zoom in on particular details.

# 4 Curvature of images $f(C_r)$ of $C_r$

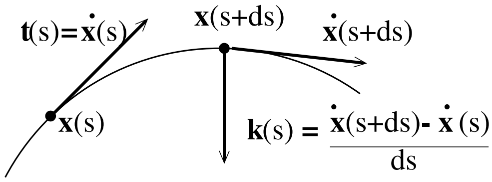


Figure 1: Coordinates and vectors associated with a curve

A smooth curve $\mathbf{x} : [a, b] \to \mathbb{C}$ is said to be *regular* if $\mathbf{x}'(t) \neq \mathbf{0}$ for all $t \in [a, b]$. It is a fact that any smooth regular curve $\mathbf{x}$ admits a smooth reparametrisation $\mathbf{x}(s)$ (where we are using $\mathbf{x}$ for both the original and reparametrised curve) such that the parameter $s$ is the distance travelled along the curve, which we regard as the "natural" parametrisation. Figure 1 shows coordinates and vectors associated with a curve such as $f(C_r)$. Here $\mathbf{x}(s)$ is the position vector of a point on the curve. The distance from $\mathbf{x}(s)$ to $\mathbf{x}(s + ds)$ is $dx = |\mathbf{x}(s + ds) - \mathbf{x}(s)|$ for an infinitesimal distance $ds$ along the curve. Hence $|\dot{\mathbf{x}}(s)| = 1$ where

$$\dot{\mathbf{x}}(s) = \lim_{ds \to 0} \frac{\mathbf{x}(s + ds) - \mathbf{x}(s)}{ds} \tag{2}$$

(this is sometimes phrased as $\mathbf{x}(s)$ is a unit speed curve). Hence the tangent vector $\mathbf{t}(s) = \dot{\mathbf{x}}(s)$ to the curve $\mathbf{x}(s)$ is always a unit vector. The vector $\mathbf{k}(s) = \ddot{\mathbf{x}}(s)$ is the curvature vector on the curve at the point $\mathbf{x}(s)$. The curvature $|\kappa|$ of the curve at the point $\mathbf{x}(s)$ is the magnitude of $\mathbf{k}(s)$,

$$|\kappa| = |\mathbf{k}(s)| \ , \tag{3}$$

and the radius of curvature $\rho$ is

$$\rho = \frac{1}{|\kappa|} = \frac{1}{|\mathbf{k}(s)|} \ . \tag{4}$$

The function $f(z)$ is not a natural representation of the curve $f(C_r)$ because neither is $z$ a scalar nor is $z_2 - z_1$ the distance in the complex plane along the curve between the two points $f(z_1)$ and $f(z_2)$. However, by using suitable coordinate transforms, an expression for the curvature vector can be found for an arbitrary parametric representation of $f(z)$.

To do so, write $f(z)$ in terms of the angle $\phi = \arg z$ to give a representation $\mathbf{x} = \mathbf{x}(\phi)$ of the curve $f(C_r)$ in terms of $\phi$:

$$x(\phi) = \mathrm{Re}\,[f(z(\phi))] = \mathrm{Re}\,[f(\phi)] \tag{5}$$
$$y(\phi) = \mathrm{Im}\,[f(z(\phi))] = \mathrm{Im}\,[f(\phi)] \ , \tag{6}$$

where $\mathbf{x}(\phi) = (x(\phi), y(\phi))$.

 

**Question 6** Show that

$$|\kappa| = \frac{|\mathbf{x}' \times \mathbf{x}''|}{|\mathbf{x}'|^3} \ , \tag{7}$$

where

$$\mathbf{x}' = \left(\mathrm{Re}\left[\frac{df(z(\phi))}{d\phi}\right],\ \mathrm{Im}\left[\frac{df(z(\phi))}{d\phi}\right]\right) \tag{8}$$

$$\mathbf{x}'' = \left(\mathrm{Re}\left[\frac{d^2 f(z(\phi))}{d\phi^2}\right],\ \mathrm{Im}\left[\frac{d^2 f(z(\phi))}{d\phi^2}\right]\right) \ . \tag{9}$$

Equation (7) gives us the magnitude but not the sign of $\kappa$. We (arbitrarily) define the sign of $\kappa$ to be positive if the curve $\mathbf{x}(s)$ is turning anticlockwise about its local centre of rotation. By expressing $\mathbf{x}'$ and $\mathbf{x}''$ in terms of $\mathbf{x}(s)$, $\mathbf{t}(s)$, $\mathbf{k}(s)$ and derivatives of $s$ with respect to $\phi$, find a way to compute the sign of $\kappa$.

Write (8) and (9) in terms of derivatives of $f(z)$ with respect to $z$.

**Programming Task:** Write a program to compute the integral

$$\kappa_{\text{tot}} = \int_{f(C_r)} \kappa \, ds \tag{10}$$

for a range of values of $r$. Use your answer to Question 6 to ensure that your program calculates the sign of $\kappa$ correctly.

**Question 7** Using your program plot a graph of $\kappa_{\text{tot}}$ against $r$ for each of the polynomials $f_1(z)$, $g(z)$ and $h(z)$. Use what you have found out in Questions 1–6 to help you to explain what you see in your graphs. For a general polynomial $f(z)$, what does $\kappa_{\text{tot}}$ tell you about the curve $f(C_r)$?

## Reference

Lipschutz, M. M., 1969: *Schaum's Outline Series: Theory and Problems of Differential Geometry*, McGraw-Hill Education – Europe.

 

# 2.4 Sensitivity of Optimisation Algorithms to Initialisation

*This project requires an understanding of the Part IA Probability and Part IA Analysis courses.*

## 1 Introduction

In modern statistics and machine learning, it is common to derive estimators and make decisions by minimising some *objective function* $f$. This task can generally not be solved in closed form. As such, a standard solution is to apply an iterative optimisation algorithm to attempt to find an approximate minimiser.

However, increasingly, statistical problems lead to complicated objective functions which admit multiple local minima. It is common practice to then run the optimisation algorithm numerous times from different initial conditions, in hope of finding the true optimum or a satisfactory one eventually. It is thus of interest to understand the sensitivity of our optimisation algorithms to their initialisation, and to understand which features of the objective function inform the outcomes of these algorithms.

## 2 Gradient Descent

The optimisation algorithm of study in this project is *gradient descent*. To run it, one must specify a differentiable *objective function* $f : \boldsymbol{D} \to \mathbf{R}$ with domain $\boldsymbol{D} \subseteq \mathbf{R}^d, d \geqslant 1$, an *initial point* $x_0 \in \boldsymbol{D}$, and a *step-size* $h > 0$. The iterates of the algorithm are then defined recursively as

$$x_t = x_{t-1} - h\nabla f(x_{t-1}) \quad \text{for } t \in \{1, 2, \ldots\}, \tag{1}$$

and the hope is that as $t \to \infty$, the iterates $x_t$ converge (numerically) to a minimiser of $f$ if $h$ is sufficiently small.

In what follows, we will focus our attention on minimisation of the 'double-well' toy function $f_\theta$, defined for $\theta \in (0, \pi)$ by

$$f_\theta : [-1, 1] \to \mathbf{R}, \quad x \mapsto \left(x^2 - \frac{3}{4}\right)^2 - x\cos\theta. \tag{2}$$

**Question 1**: Find the stationary points of this function in analytic form, and classify them as local minima, maxima, or saddlepoints. [*Hint: the stationary points can all be expressed as trigonometric functions; in particular, in your calculations you may use an expression for the cosine of the triple angle.*]

Given that we know the analytic form of the stationary points of $f_\theta$, we can easily evaluate the performance of gradient descent (or indeed, any other optimisation algorithm).

**Question 2**: Take $\theta = \frac{\pi}{6}, h = 0.01$, and run gradient descent on $f = f_\theta$ for 1000 steps, from initial points $x_0 \in \{\frac{k}{50}\}_{k=-50}^{50}$. What do you observe about the outcomes?

 

## 3 The Monte Carlo Method

In computational settings, it is often the case that a quantity of interest $\nu$ is naturally expressed as an expectation, i.e. there is some random variable $X$ and some function $g$ such that

$$\nu = \mathbf{E}\left[g(X)\right]. \tag{3}$$

The *Monte Carlo method* (MC) involves drawing $N$ independent samples $\{X^i\}_{i=1}^N$ which are distributed like $X$, and forming the estimator

$$\hat{\nu}_N \triangleq \frac{1}{N}\sum_{i=1}^{N} g(X^i). \tag{4}$$

**Question 3**: Show that $\hat{\nu}_N$ is *unbiased* for $\nu$, i.e. $\mathbf{E}\left[\hat{\nu}_N\right] = \nu$. Assuming that $\mathbf{Var}(g(X)) < \infty$, obtain an expression for the variance of $\hat{\nu}_N$.

Returning to our toy function above, fix $\theta \in (0, \pi)$, $f = f_\theta$, and let $\{X_t^h\}_{t=0,1,...}$ be the sequence of random variables obtained by i) sampling an initial point $X_0^h \sim \text{Uniform}([-1, 1])$, and ii) iterating $X_t^h = X_{t-1}^h - h\nabla f(X_{t-1}^h)$.

For some $T, h > 0$ such that $Th^{-1} \in \boldsymbol{N}$, we are interested in studying the behaviour of

$$\mu^h \triangleq \mathbf{E}\left[X_{Th^{-1}}^h\right] \qquad \text{and} \qquad \mu \triangleq \lim_{h\to 0^+} \mu^h, \tag{5}$$

i.e. the outcome of running gradient descent from a *randomised* initial point, using smaller and smaller step-sizes, and run for longer and longer. We take $T = 10$ fixed throughout, as in this example, this is approximately sufficient for convergence to take place.

A basic approach to estimating $\mu$ is to take $h$ as small as possible, and to estimate $\mu^h$ as accurately as possible by the Monte Carlo estimate $\hat{\mu}_N^h$ by taking $N$ as large as possible.

**Question 4**: Test this method out: fix $\theta = \frac{\pi}{4}$, and for $k \in \{0, 1, \cdots, 10\}$, take $h = 0.1 \cdot 2^{-k}$, and estimate $\mu^h$ using $N_k = 2^{20-k}$ samples, so that the same amount of computational time is used for each $k$. What do your estimates suggest about the behaviour of $\mu^h$ as $h$ decreases? What do they suggest about the variance of $X_{Th^{-1}}^h$ as $h$ decreases?

In this approach, because $h$ is not exactly 0, we incur a finite *bias*, i.e. even in the limit of infinitely many samples, our estimator would converge to $\mu^h \neq \mu$. As such, the variance of our estimator would not fully reflect its accuracy. Instead, it is standard to use the following more general measure of accuracy. The *mean squared error* (MSE) of an estimator $T$ of a quantity $\tau$ is defined by

$$\mathbf{MSE}(T; \tau) \triangleq \mathbf{E}[(T - \tau)^2]. \tag{6}$$

**Question 5**: Prove the 'bias-variance decomposition', i.e. show that

$$\mathbf{MSE}(T; \tau) = (\mathbf{E}[T] - \tau)^2 + \mathbf{Var}(T). \tag{7}$$

We present the following facts without proof: for $h$ sufficiently small, there are constants $A_1, A_2, A_3 \in (0, \infty)$ such that

1. the bias of the approximation $\mu^h$ is bounded as $|\mu^h - \mu| \leqslant A_1 h$,
2. the variance of $X_{Th^{-1}}^h$ is bounded as $\mathbf{Var}(X_{Th^{-1}}^h) \leqslant A_2$, and
3. for $t \in \{0, 1, \ldots\}$, the cost of generating a sample of $X_t^h$ satisfies $\mathbf{Cost}(X_t^h) = A_3 t$.

 

**Question 6**: Suppose we estimate $\mu$ by fixing $h > 0, N \in \mathbf{N}$, generating $N$ i.i.d. samples $\{Y^i\}_{i=1}^N$ distributed as $X^h_{Th^{-1}}$, and forming the estimator

$$\hat{\mu}^h_N = \frac{1}{N} \sum_{i=1}^{N} Y^i. \tag{8}$$

Use the bias-variance decomposition to show that the MSE of $\hat{\mu}^h_N$ can be bounded above by $A_1^2 h^2 + \frac{A_2}{N}$. Suppose now that the computational budget is $C$, i.e. the cost of generating all of the random variables used in the MC estimator is bounded above by $C$. Assume that we use our full budget, i.e. we choose $(N, h)$ such that $N \cdot \frac{A_3 T}{h} = C$. Use this to express the upper bound on the MSE as a function of only $h$, and derive the $h$ which minimises this upper bound. How does the optimal MSE scale with $C$?

## 4 Multi-Level Monte Carlo

For a given computational budget $C$, it is possible construct estimators with less variability than $\hat{\mu}^h_N$, and hence improve our accuracy. We exploit the intuition that if the initial point $x_0$ is fixed, we expect that the paths of $X^h_t$ and $X^{h/2}_{2t}$ will stay close together, and thus that $\mu^h \approx \mu^{h/2}$.

In order to justify this later on, we introduce an extra fact without proof: for $h$ sufficiently small, there is a constant $A_4 \in (0, \infty)$ such that

4. if two sequences of gradient descent iterates have the same initial point $X_0 \sim \text{Uniform}([-1, 1])$, then $\mathbf{Var}\left(X^h_{Th^{-1}} - X^{h/2}_{2Th^{-1}}\right) \leqslant A_4 h^2$.

We note quickly that the facts presented before Question 6 remain true in what follows.

For $X_0 \sim \text{Uniform}([-1, 1])$ and $l = 0, \ldots, L$, $L \in \mathbf{N}$, define $h_l = 0.1 \times 2^{-l}$, let $X^{h_l}_{Th_l^{-1}}$ be the $(Th_l^{-1})^{\text{th}}$ gradient descent iteration for $f_\theta$, with $X^{h_l}_0 = X_0$, and with step-size $h_l$. Define the random variables

$$Y_0 = X^{h_0}_{Th_0^{-1}} \qquad \text{and} \qquad Y_l = X^{h_l}_{Th_l^{-1}} - X^{h_{l-1}}_{Th_{l-1}^{-1}}, \quad l = 1, \ldots, L. \tag{9}$$

We can then formally write that

$$\mu = \sum_{l \geqslant 0} \mathbf{E}\left[Y_l\right]. \tag{10}$$

**Question 7**: Justify that the above sum converges absolutely, and find an upper bound for the truncation error incurred by approximating $\mu \approx \sum_{l=0}^{L} \mathbf{E}\left[Y_l\right]$.

With this in mind, we can aim to approximate $\mu$ by taking a *truncation level* $L$, a sequence of *level sizes* $\{N_l\}_{l=0}^L$, and forming the *Multi-Level Monte Carlo estimator* (MLMC)

$$\hat{\mu}_{N_{1:L}} \triangleq \sum_{l=0}^{L} \left[\frac{1}{N_l} \sum_{i=1}^{N_l} Y^i_l\right]. \tag{11}$$

where for each $i$, $\{Y^i_l\}_{i=1}^{N_l}$ are independent, identically-distributed (iid) samples of $Y_l$, i.e., for each $(i, l)$ we independently draw an initial point $X_0^{(i,l)} \sim \text{Uniform}([-1, 1])$ and define $Y^i_l$ as in display (9) using $X_0^{(i,l)}$ rather than $X_0$. Hence, $\{Y^i_l\}_{i,l}$ are mutually independent.

 

**Question 8**: For $\theta \in \left\{\frac{k\cdot\pi}{2^7}\right\}_{k=1}^{2^6}$, compute $\hat{\mu}_{N_{1:L}}$, taking $L = 10$ and using level sizes i) $N_l \equiv 5$ and ii) $N_l = 2^{L-l}$. Which estimator exhibits greater variability?

**Question 9**: Derive an upper bound for the MSE of the MLMC estimator with truncation level $L$ and level sizes $\{N_l\}_{l=0}^{L}$.

We now try to choose $L$ and $\{N_l\}_{l=0}^{L}$ such that the MSE of the resulting MLMC estimator is minimised given a fixed computational budget.

**Question 10**: Suppose that the computational budget is $C$, i.e. the cost of generating all of the random variables used in the MLMC estimator is bounded above by $C$. By treating the $N_l$ as continuous variables, $L = \infty$, derive an allocation of level sizes $\{\tilde{N}_l\}_{l\geqslant 0}$ which minimises the upper bound for the MSE of the resulting MLMC estimator derived in question 9. [*Hint: To minimise a function $F(x)$ subject to the constraint that $G(x) = c$, it suffices to identify stationary points of $H(x, \lambda) = F(x) + \lambda(G(x) - c)$. This is known as the method of Lagrange multipliers.*]

From now on, we move back to integer-valued levels by taking $N_l = \lfloor \tilde{N}_l \rfloor$.

**Question 11**: How do you find that $L$ scales with $C$? Derive an expression for how the optimal MSE scales as $C$ grows, and compare this to the MC estimator from Question 5.

## 5 Application to Double-Well Loss Function

We will now use the estimators derived above to study the behaviour of gradient descent on the double-well function $f_\theta$ defined earlier.

**Question 12**: Define $m_1(\theta)$ and $m_2(\theta)$ as the local *minima* of $f_\theta$ in $[-1, 1]$, defined such that $m_1(\theta) < m_2(\theta)$. Suppose that $h > 0$ and $T \in \mathbf{N}$ are sufficiently small and large, respectively, so that $\min\{|m_1(\theta) - X_T^h|, |m_2(\theta) - X_T^h|\} \approx 0$ for any initial point $x_0 \in [-1, 1]$. Define

$$p_1(\theta) = \mathbf{P}\left(\lim_{T\to\infty} X_T^h = m_1(\theta)\right) \qquad \text{and} \qquad p_2(\theta) = \mathbf{P}\left(\lim_{T\to\infty} X_T^h = m_2(\theta)\right). \tag{12}$$

Derive an expression for $p_1(\theta)$ and $p_2(\theta)$ in terms of $\mu, m_1(\theta)$ and $m_2(\theta)$.

**Question 13**: Use a Multi-Level Monte Carlo scheme with the optimal level sizes, as derived in Question 9, to form estimates of $p_1(\theta)$ and $p_2(\theta)$ for $\theta \in \left\{\frac{k\cdot\pi}{2^7}\right\}_{k=1}^{2^6}$. Plot your estimates to show how they vary with $\theta$. For which values of $\theta$ does the outcome of running gradient descent on $f_\theta$ vary most?

 