Introduction to numerical projects

Here follows a brief recipe and recommendation on how to write a report for each project.

- Give a short description of the nature of the problem and the eventual numerical methods you have used.
- Describe the algorithm you have used and/or developed. Here you may find it convenient to use pseudocoding. In many cases you can describe the algorithm in the program itself.
- Include the source code of your program. Comment your program properly.
- If possible, try to find analytic solutions, or known limits in order to test your program when developing the code.
- Include your results either in figure form or in a table. Remember to label your results. All tables and figures should have relevant captions and labels on the axes.
- Try to evaluate the reliabilty and numerical stability/precision of your results. If possible, include a qualitative and/or quantitative discussion of the numerical stability, eventual loss of precision etc.
- Try to give an interpretation of you results in your answers to the problems.
- Critique: if possible include your comments and reflections about the exercise, whether you felt you learnt something, ideas for improvements and other thoughts you've made when solving the exercise. We wish to keep this course at the interactive level and your comments can help us improve it.
- Try to establish a practice where you log your work at the computerlab. You may find such a logbook very handy at later stages in your work, especially when you don't properly remember what a previous test version of your program did. Here you could also record the time spent on solving the exercise, various algorithms you may have tested or other topics which you feel worthy of mentioning.

Format for electronic delivery of report and programs

The preferred format for the report is a PDF file. You can also use DOC or postscript formats. As programming language we prefer that you choose between C/C++, Fortran or Python. The following prescription should be followed when preparing the report:

- Use Fronter to hand in your projects, log in at blyant.uio.no and choose 'fellesrom fys3150'. Thereafter you will see an icon to the left with 'hand in' or 'innlevering'. Click on that icon and go to the given project. There you can load up the files within the deadline.

- Upload **only** the report file and the source code file(s) you have developed. The report file should include all of your discussions and a list of the codes you have developed. Do not include library files which are available at the course homepage, unless you have made specific changes to them.
- Comments from us on your projects, approval or not, corrections to be made etc can be found under your Fronter domain and are only visible to you and the teachers of the course.

Finally, we do prefer that you work two and two together. Optimal working groups consist of 2-3 students. You can then hand in a common report.

Project 4, Monte Carlo simulations of financial transactions, deadline November 11

The aim of this project is to simulate financial transactions among financial agents using Monte Carlo methods. The final goal is to extract a distribution of income as function of the income m. From Pareto's work (V. Pareto, 1897) it is known from empirical studies that the higher end of the distribution of money follows a distribution

$$w_m \propto m^{-1-\alpha},$$

with $\alpha \in [1,2]$. We will here follow the analysis made by Patriarca *et al*, see Physica A**340**, 334 (2004).

Here we will study numerically the relation between the micro-dynamic relations among financial agents and the resulting macroscopic money distribution.

We assume we have N agents that exchange money in pairs (i,j). We assume also that all agents start with the same amount of money $m_0 > 0$. At a given 'time step', we choose randomly a pair of agents (i,j) and let a transaction take place. This means that agent i's money m_i changes to m_i' and similarly we have $m_j \rightarrow m_j'$. Money is conserved during a transaction, meaning that

$$m_i + m_j = m_i' + m_j'. \tag{1}$$

The change is done via a random reassignement (a random number) ϵ, meaning that

$$m_i' = \epsilon(m_i + m_j),$$

leading to

$$m_j' = (1-\epsilon)(m_i + m_j).$$

The number ϵ is extracted from a uniform distribution. In this simple model, no agents are left with a debt, that is $m \geq 0$. Due to the conservation law above, one can show that the system relaxes toward an equilibrium state given by a Gibbs distribution

$$w_m = \beta \exp{(-\beta m)},$$

with

$$\beta = \frac{1}{\langle m \rangle},$$

and $\langle m \rangle = \sum_i m_i/N = m_0$, the average money. It means that after equilibrium has been reached that the majority of agents is left with a small number of money, while the number of richest agents, those with m larger than a specific value m', exponentially decreases with m'.

We assume that we have $N = 500$ agents. In each simulation, we need a sufficiently large number of transactions, say 10^7. Our aim is find the final equilibrium distribution w_m. In order to do that we would need several runs of the above simulations, at least $10^3 - 10^4$ runs (experiments).

a) Your task is to first set up an algorithm which simulates the above transactions with an initial amount m_0. The challenge here is to figure out a Monte Carlo simulation based on the above equations. You will in particular need to make an algorithm which sets up a histogram as function of m. This histogram contains the number of times a value m is registered and represents $w_m \Delta m$. You will need to set up a value for the interval Δm (typically $0.01 - 0.05$). That means you need to account for the number of times you register an income in the interval $m, m + \Delta m$. The number of times you register this income, represents the value that enters the histogram. You will also need to find a criterion for when the equilibrium situation has been reached.

b) Make thereafter a plot of $\log(w_m)$ as function of m and see if you get a straight line. Comment the result.

c) We can then change our model to allow for a saving criterion, meaning that the agents save a fraction λ of the money they have before the transaction is made. The final distribution will then no longer be given by Gibbs distribution. It could also include a taxation on financial transactions.

   The conservation law of Eq. (1) holds, but the money to be shared in a transaction between agent i and agent j is now $(1-\lambda)(m_i + m_j)$. This means that we have

$$m_i' = \lambda m_i + \epsilon(1-\lambda)(m_i + m_j),$$

   and

$$m_j' = \lambda m_j + (1-\epsilon)(1-\lambda)(m_i + m_j),$$

   which can be written as

$$m_i' = m_i + \delta m$$

   and

$$m_j' = m_j - \delta m,$$

   with

$$\delta m = (1-\lambda)(\epsilon m_j - (1-\epsilon) m_i),$$

showing how money is conserved during a transaction. Select values of $\lambda = 0.25, 0.5$ and $\lambda = 0.9$ and try to extract the corresponding equilibrium distributions and compare these with the Gibbs distribution. Comment your results. If you have time, see if you can extract a parametrization of the above curves (see the article of Patriarca *et al*, Physica A**340**, 334 (2004).)