# CF963 Computational Models in Economics and Finance

Assignment, 2020/21

- Answer all (four) questions below. Submit your answers to FASER. You need to submit:
  - a single report with your answers to all questions,
  - all MATLAB files that you created in the context of this assignment.

  Submit them separately (do NOT .zip). Make sure that your code is easy to follow, and copy and paste your code in the report as requested in the specific tasks.
- Your assignment will be assessed on the quality of the files you submit –correctness, work quality and quality of presentation. Aim for precise and concise answers and explanations. Good luck!

## Task 1 [20%]

Consider the following moving average trading strategy:

Let $7^{MA}$ be the 7-days moving average, and let $14^{MA}$ be the 14-days moving average. If the $7^{MA}$ crosses the $14^{MA}$ from below, then buy your entire budget. If the $7^{MA}$ crosses the $14^{MA}$ from above, then sell your entire portfolio.

Your task is to:

a. (10%) implement the above strategy and test it with MATLAB on the JET.L (JustEat stock 6M) stock's daily closing prices that are provided on the moodle page of the module (Unit 1). Assume you have £1M available to invest.

The output of your code should include the following (no particular format is required, as long as ithe requested information is clearly presented):

b. (4%) when your algorithm buys or sells,

c. (4%) how much your algorithm buys or sells in each deal,

d. (2%) what profit/loss your algorithm made in total.

Copy and paste your code and present the requested outputs in the report (in addition to uploading the matlab file on FASER).

## Task 2 [30%]

Program the agent-based simulator in MATLAB for the following setting:

There are 20 buyers and 20 sellers and each seller initially owns 30 units of the item that is being traded in a double auction market. Each buyer $i \in \{1, \dots, 20\}$, has a fixed valuation $v_i$ that is a random number between 1 and 200 (such random selections have to be implemented by your code); valuations don't change between different rounds of the simulation. At every round, buyers and sellers place their orders that are of the following form: A buyer's order $(b, q)$ is such that

- $b$ (bid) is the maximum amount the corresponding buyer would be willing to spend for one unit of the item and is a random positive integer number that is at most equal to the valuation of the buyer. In other words, the bid is selected uniformly at random from the range $[1, \dots, v_i]$), and
- $q$ (quantity) is the number of units the corresponding buyer would like to buy at this round (for the given price) and is a random integer number between 1 and 5.

Similarly, a seller's order $a$ is such that

- $a$ (ask) is the minimum amount of money the corresponding seller would be willing to get for one unit of the item at this round of the simulation and is is a random integer number that is at least equal to the number corresponding to 10 times the round number, and at most equal to 200.

After all orders have been placed in each round, the maximum number of trades is being implemented while giving priority to high bids and low asks. Note that a buyer's order can be matched to more than one sellers, e.g. if two sellers only have one remaining unit of the item each and a buyer wishes to buy two units. A trade can be performed if the bid is at least as high as the ask. Once a seller sells their 30 units they are no longer considered for trades.

a. (15%) Run your simulation for 10 rounds and:

b. (5%) Calculate the spread (difference) between the best bid and the best ask at each round and plot it.

c. (5%) Compute how many units of the item were traded in total, and

d. (5%) Compute the number of units that are available for sale for each possible price after the final round.

Copy and paste your code and present the requested outputs, including the plot, in the report (in addition to uploading the matlab file on FASER). No particular format is required, as long as ithe requested information is clearly presented. Marks will be awarded for partial answers.

## Task 3 [20%]

a. (13%) Consider the Cournot duopoly model where the inverse demand function and the cost functions are given by

$$P = 120 - Q, \qquad c_1 = 10 + 3q_1, \qquad c_2 = 12 + 6q_2,$$

where $Q = q_1 + q_2$ is the total production quantity and $q_i$ is the production quantity of firm $i$, for $i = 1, 2$. Give the profit functions of the firms and compute the Nash equilibrium defined by the quantity each firm chooses to produce. Compute the profit of each firm, the consumer surplus, and the total surplus at equilibrium.

b. (7%) Consider the leader-follower duopoly model with the inverse demand function and the cost functions as defined in Part a. Let the reaction function of firm 2 be

$$r_2(q_1) = 57 - \frac{q_1}{2}.$$

Give the profit function of firm 1 and find the equilibrium strategies (production quantities) of the firms.

## Task 4 [30%]

Pick one of the following papers and provide

a. (10%) a concise summary that reveals the main points addressed in the paper, and

b. (20%) a critical assessment of the paper.

Focus on the real life setting that is considered, the modelling choices that were made in an attempt to abstract it and analyse it, and elaborate on the particular computational modelling technique that is applied to it.

Suggestions for points to address: What simplifying assumptions are made? How does computational thinking help us analyze this particular situation? Are the assumptions made and/or the methodology used appropriate? How could this analysis be extended, e.g. can you think of an adaptation to the model that would be meaningful?

Length guide: Your answer should not exceed an A4 page overall. Aim for half page summary of the paper and another half page for criticism on the approach. There is no need to focus on the technical details (mathematical proofs).

Papers:

*Complexity of Stability in Trading Networks*, by Tamás Fleiner, Zsuzsanna Jankó, Ildikó Schlotter, Alexander Teytelboym. [link]

*Personal Finance Decisions with Untruthful Advisors: an Agent-Based Model*, by Loretta Mastroeni, Maurizio Naldi, and Pierluigi Vellucci. [link]

*FairLedger: A Fair Blockchain Protocol for Financial Institutions*, by Kfir Lev-Ari, Alexander Spiegelman, Idit Keidar, Dahlia Malkhi. [link]

*Deep Reinforcement Learning for Trading*, by Zihao Zhang, Stefan Zohren, Stephen Roberts. [link]

## Please refer to the Student's handbook on the Departmental Policy on Plagiarism and Late Submission