IFT 6521 PROGRAMMATION DYNAMIQUE

EMMA FREJINGER
frejinge@iro.umontreal.ca

PROJECT

The aim of this project is to apply the methodologies acquired during the course to solve a stochastic sequential decision problem of your choice and analyze the results. The problem definition can be inspired from existing work, but it has to be your own creation. The instructions are divided into two variants :

- *"Minimum" : variant which can give a maximum grade of B +*
- *"Advanced" : variant which can give a maximum grade of A +*

Hand in and compulsory attendance. The final hand in of the project must be made no later than April 8 at 8 :00 am. Your presence during the lecture April 8 is compulsory. The hand in must be made on StudiUM. All files (report and code) must be compressed into a single zip file with the name : LastName_First Name_MinimumOUAvance.zip (the last part refers to the variant of the project you have chosen).

1 Report and instructions

Here are the sections (1–8) that should be in your report, with instructions on the content of each :

1. Description of the problem. Describe all the elements of the stochastic sequential decision problem chosen without mathematical notation and without stating the modeling assumptions. You can take inspiration from existing problems (courses, internet, books, ...) but you must create your own variant. Don't forget to cite your sources of inspiration. This section answers the following questions : What is the problem ? In which application (or context) is it important ? Maximum 1.5 pages.

2. Modeling. Describe the dynamic programming formulation. That is, introduce the mathematical notation and justify your modelling hypotheses. Clearly define states, actions, transition functions, costs, and recurrence equations for your problem specifically. Maximum 1.5 pages.

3. Algorithms. Describe the algorithms you have chosen to implement and motivate your choices. The description must refer to equations and / or pseudo-codes. In the results section, you need to compare the performance of an exact method with at least one approximate method. It is therefore necessary to describe at least two algorithms. You must also describe how you evaluate the expectation in the recursion. The requirement of approximate method varies according to the variant of the project. Maximum 2 pages.

Choice of approximate method :

— "Minimum" variant : at least one rollout variant

— "Advanced" variant : at least one variant of parametric approximation

Note : you have several modelling choices to do which is not detailed here, for example, choice of finite or infinite horizon. You must implement two methods (exact and approximate) taught in this course. If you want to implement additional algorithms in order to make comparisons, you can choose more freely. For example, you can implement reinforcement learning algorithms if you like.

4. Results and analysis. This section describes your instances, the motivation and structure of your experimental study, the results and the corresponding analysis. You must include at least three instances :

— Small : this instance serves as an illustrative example. It must give an intuition on the problem and convince the reader that your modeling and your algorithms are correct.

— Medium and large : these instances are used to compare the exact method with the approximate method. Instances should be chosen to illustrate the differences between the methods.

You have to choose wisely how to present solutions and performance measures to make your analysis clear and convincing. If you get counterintuitive results, be sure to mention it explicitly, giving possible reasons. Maximum 3 pages.

5. Conclusion. This section is a summary of your work and should overview the key takeaways. The second part is a discussion that links your problem to methods taught in the course that you have not used in your work. More specifically, you must motivate why methods / models (approximate dynamic programming, infinite horizon) seen in the courses after the reading week would or would not apply to your problem. Maximum 0.5 page for the first part and maximum 1.5 pages for the second part.Maximum 0.5 page.

6. Instructions. Explain how to run your code to generate the results given in the results section. Briefly describe the structure of the code in relation to the content

in Section 3 of the report. If applicable, make sure you have clearly identified the beginning and the end as well as the source of each piece of code that you did not write yourself. Only code that you have written yourself will be assessed and graded in the course. Of course, you have to write the code related to the algorithms yourself. The code must be submitted with the report file. You can choose your programming language from Java, C, C ++, Python, Julia and Matlab. If you want to choose another language, ask permission first. Maximum 1 page.

7. References. List the references you have consulted (websites, books, articles, etc.). The references have to be complete. For example, just including a link to a webpage is not sufficient. If you do not know how to write a bibliography, make sure to find out.

8. Self-assessment. For each element in the assessment grid (see the GridEvaluationProject.pdf file on StudiUM), estimate your score with a short motivation. This is a way to verify that you have all the parts and to be critical of your own work. If you want, you can discuss your self-assessment with another student. Obviously, the grade given by the professor may be different from your estimate - it may be better or worse...

Presentation – only "advanced" variant Prepare a short presentation that summarizes the work (Sections 1 - 5 and 7 of the report). During the last lecture, the students haven chosen this variant of the project will present their work. The duration of the presentation will depend on the number of students (approximately 10 minutes maximum). You will receive instructions for the process separately.

2 Evaluation criteria

The final grade for the course is entirely based on the final project, but to have a passing grade you must also have successfully completed the assignments. The GridEvaluationProject.pdf file contains the detailed evaluation criteria. Read them carefully before starting your project.