

# Stanford CME 241 Winter 2026

Course Project Guidance by Prof. Ashwin Rao

## 1 Introduction

My goal here is to provide guidance on a broad topic for the course project that is of high practical relevance, has traditionally been challenging to solve, and is well suited for appropriate formulation and solution using the mathematical and computational techniques taught in this course. Each project group can specialize the project scope within this broad topic.

The broad topic is **Personal Finance Optimization**. At various stages of our lives, we are trying to grow our wealth while simultaneously making payments on our obligations and tactically paying for optional/selective products and services that give us specific forms of satisfaction or enjoyment. We typically grow our wealth by receiving salary, bonuses and equity while doing jobs and by making appropriate choices on investments made with our savings (eg: stocks, real-estate, gold etc.) Our payments obligations are typically our rent or mortgage, vehicle payments, utility bills etc. Our tactical payments for optional/selective products/services giving us satisfaction/enjoyment can be wide-ranging - a vacation, a luxury car, a birthday celebration, a gift for a loved one etc. The objective is to maximize the aggregated *Utility of Consumption* (something that you will learn about in detail in Chapter 7 of the [textbook](#), also covered in a lecture during this course). The problem is to make a sequence of optimal decisions on the optional/selective products/services, amidst significant uncertainty with how our lives will transpire.

## 2 Choosing a Problem Specialization

Your goal is to identify an appropriate problem specialization within this broad topic of *Personal Finance Optimization* - a specialization that is practically relevant and yet tractable with the mathematical and computational tools within the space of Stochastic Control, and particularly using RL techniques.

In Chapter 8 of the [textbook](#), which I will cover in detail in a couple of lectures during this course, you will learn about a few problem specializations within this broad topic. In this Chapter, the focus is on *Optimizing Retirement Plans*, where one balances between investments (to grow money) and consumption (to enjoy retirement life). You can surely choose a practical version of Retirement Planning for this project, but you can also consider personal finance problems for younger people during their careers (where there are the additional choices of which career to pursue, which job to choose at any point, how to position for growth on the job etc.).

As you would note, some problem specializations can get very complicated and sometimes the decision choices can be hard to formalize (eg: career decisions). A key aspect of this project is to make a good choice on the right problem specialization to pick for this course project, bearing in mind that you could continue this project after the course, potentially leading to a commercial application, and perhaps yielding the foundation of a B2C startup for you to pursue (eg: a retirement optimization app for senior citizens).

## 3 Project Phases and Grading

Since we do not have a midterm or final exam in this edition of CME 241, the project forms a significant portion of your grade (50%) and you will start on the project from a fairly early stage in this course. The project is divided into 3 phases which respectively involve problem definition, solution of a simplified version of the problem using Dynamic Programming, and finally the main phase where you will solve a more realistic version of the problem by creating a simulated environment and a Reinforcement Learning algorithm. Below I detail the expectations from these 3 project phases.

### 3.1 Phase 1: Problem Definition - 10 Points

This phase carries 10 points credit for the course and the due date is February 9. The expectation here is to first describe the problem specialization and its nuances as well as practical relevance, then to write the Markov Decision Process for the problem specialization in clear mathematical notation as well as code specification as a *MarkovDecisionProcess* class. You have to define 3 incrementally diluted versions of the problem, as follows:

1. This is the version of the problem you would ideally like to solve if you had ample time and resources, something that would be practically useful and would serve as the technology for a commercial product.
2. This is the diluted version of the above problem that you can solve by the end of this course (in Phase 3).
3. This is the further diluted version of the above problem that you can solve in Phase 2 based on everything you learn in the first 4 weeks, i.e., before you learn RL.

The key is to parameterize the problem appropriately in a *MarkovDecisionProcess* class so that you can solve it with say Dynamic Programming (DP)/Approximate Dynamic Programming (ADP) for an appropriate simplified setting of the parameters, and then reuse the same *MarkovDecisionProcess* class with a more ambitious setting of the parameters so that you can then solve with RL.

### 3.2 Phase 2: Solving a Simplified Version of the Problem (before you learn RL techniques) - 15 points

This phase carries 15 points credit for the course and the due date is February 23. The idea is to use DP/ADP to solve a version of the problem without too large a state space/action space that can run and converge in a reasonable amount of time. Ideally, in this phase, you will learn a lot about the tradeoff between capturing relevant practical details of the problem versus keeping things simple/small to enable tractability for DP/ADP. You will be able to experience why RL is a necessity to overcome the Curse of Dimensionality and the Curse of Modeling.

### 3.3 Phase 3: Solving a Realistic Version of the Problem with RL - 25 points

This phase carries 25 points credit for the course and the due date is March 16. We will have project presentations on March 13 but the finished submission of the project (with code and paper) is due March 16. The idea is to first build a simulated environment by generating sampling traces from the *MarkovDecisionProcess* you had set up in Phase 1, and being able to generate those sampling

traces for a realistic version of the problem (with appropriate parameterization of the *MarkovDecisionProcess*). Once the simulated environment is available through these sampling traces, you will identify the appropriate RL algorithm to solve the problem (ideally, you will modify a standard RL algorithm to fit the specific nature of the problem you have here). Since you only have a few weeks for this phase, you will need to make sure you don't take on something that is hard to solve in this limited time (the full blown practical version of the problem can be done by extending your work after the course, and hopefully turned into a commercial product). At the same time, you need to make sure the version you are solving in this phase is reasonably realistic (not too simplistic). You can approach simpler parameterizations of the problem with the RL code I have developed for this course (to validate your approach), but I would encourage you to use open-source RL libraries that are built for performance and scale (the RL code I have developed for this course is meant for educational purposes and hence not optimized for performance and scale).

## 4 Use of LLMs

I highly encourage you to use the full power of LLMs throughout this project. Here are some ways where LLMs will be really useful in this project.

1. Describe your goals in this project clearly to the LLM, and ask it to brainstorm with you the exact problem specialization that would be ideal for you to pursue
2. Ask the LLM to teach you about practical nuances of the problem. For example, if you take on the retirement planning problem, LLMs will do great research on this topic for you and bring back practical considerations like taxation, retirement account constraints, bequeth considerations etc., things you might not be familiar with since you are not anywhere close to retirement yourself!)
3. Vide coding and in-editor completions to accelerate your coding work
4. Identifying a good open-source RL library for you to use that will help you with performance and scaling
5. Helping you write the final project report and putting together a good final presentation

My hope is that LLMs will give you the ability to tackle a fairly ambitious version of the problem in this course. I will warn you though that you should take everything that LLMs give you with a healthy dose of doubt on the precision of its response. If you copy-paste everything you get from LLMs, it will lead you down a failure path. If you understand, probe deep, build upon and appropriately modify what you get back (based on your depth of understanding of MDPs and RL), always staying in control of all the technical work, you stand to gain significantly on this project by leveraging LLMs.