OR Programming Project

Jordi Villà-Freixa

2025-10-01

Operations Research Final Project Information

- Project Topics
  - Notes on a few specific types of projects
- Project Submission Phases
  - Evaluation
  - 1st submission: Project Proposals
    - Format
    - Grading
  - 2nd submission: Milestone
    - Contributions
    - Grading
    - Format
  - Final Writeup
    - Format
    - Contributions
    - Code
    - Grading
- After the Course
- Project FAQs

One of the main goals of this course is to prepare you to apply operations research (OR) methods to real-world tasks, or to leave you well-qualified to start research or work in optimization and decision sciences. The final project is intended to start you in these directions.

Please first have a look through the Project FAQs.

Project Topics

Choose a project type:

1. **Application:** Apply OR methods to a real-world problem (e.g., scheduling, logistics, resource allocation, network flows).
2. **Algorithmic:** Develop or improve an optimization algorithm for a specific problem.

3. **Theoretical:** Prove new or interesting properties of an OR method or algorithm.

Projects may combine these elements. Select a topic you find interesting and ambitious. For inspiration, review existing research (e.g., Google Scholar). You can also check find interesting hints here.

Identify relevant problem instances early. If data requires significant preprocessing or collection, plan accordingly —methodology and results discussion are expected. A very good OR project will be a publishable or nearly-publishable piece of work.

Notes on a few specific types of projects

- **Optimization projects:** Since this course discusses many concepts in optimization, we ask that if you decide to work on a computational project, please make sure that you use other material you learned in the class as well. For example, you might set up linear programming and integer programming baselines, or do some analysis using network flow methods. We prioritize methodological and experimental rigor and standardize our grading, so note that pursuing (or not pursuing) a computational project will not itself impact your grade. Finally, solving large optimization problems can be very time consuming, so make sure you have the necessary computing resources. Ask me if you need additional resources.
- **Replicating results:** Replicating the results in a paper can be a good way to learn. However, we ask that instead of just replicating a paper, also try using the technique on another application, or do some analysis of how each component of the model or algorithm contributes to final performance. In other words, your project does not need to be completely novel, but should not just duplicate previous work done by others.

Project Submission Phases

Your project will be submitted in three phases:

1. **1st submission: Proposal**
   Submit a brief (one page) proposal outlining your project topic, motivation, planned methods, and expected outcomes.
2. **2nd submission: Milestone**
   Submit an intermediate, functional, and testable implementation of your code, along with a draft of your report. This should demonstrate progress and allow for feedback.
3. **Final Submission: Full project**
   Submit the complete project, including all code, results, and the final report in the form of a scientific article.

Evaluation

Projects will be evaluated based on:

- **Technical quality:** Does the technical material make sense? Are the things tried reasonable? Are the proposed algorithms or applications clever and interesting? Do the authors convey novel insight about the problem and/or algorithms? Does the project have sufficient scope for the given team size?
- **Originality:** Did the authors add their own methods or analysis? Does the final project avoid being a mirror image of existing papers/projects with no net new work?
- **Communication:** Are the authors able to clearly and effectively explain the work that they did, including context, methods, and results? Are references useful to understand the scope of the article? Do the paper balance clarity with rigor? Is the document well structured?

In order to highlight these components, it is important you present a solid discussion regarding the learnings from the development of your method, and summarizing how your work compares to existing approaches.

1st submission: Project Proposals

In the project proposal, you'll pick a project idea to work on early and receive feedback from the professor. In the proposal, below your project title, include the project category (which unit of the syllabus the project is mostly linked to).

Format Your proposal should be a PDF document, giving the title of the project, the project category, the full names of all of your team members, and a 300-500 word description of what you plan to do.

Your project proposal should include the following information:

- **Motivation:** What problem are you tackling? Is this an application or a theoretical result?
- **Method:** What OR techniques are you planning to apply or improve upon?
- **Intended experiments:** What experiments or computational studies are you planning to run? How do you plan to evaluate your OR method or algorithm?

Presenting pointers to one relevant dataset/problem instance and one example of prior research on the topic are a valuable (optional) addition.

- I suggest you use LaTeX as a good tool to produce a high quality report of your work. In particular, you can use any template you can find in overleaf (e.g., this one) to easily work in collaboration within the group to write the report.

- A github repository will be needed to submit the code. Ideally, the github repo should also contain the manuscript LaTeX files.

Grading The project proposal is mainly intended to make sure you decide on a project topic and get feedback from the profesor and from your class peers. As long as your proposal follows the instructions above and the project seems to have been thought out with a reasonable plan, you should do well on the proposal.

2nd submission: Milestone

The milestone will help you make sure you're on track, and should describe what you've accomplished so far, and very briefly say what else you plan to do. You should write it as if it's an "early draft" of what will turn into your final project. You can write it as if you're writing the first few pages of your final project report, so that you can re-use most of the milestone text in your final report.

Please write the milestone (and final report) keeping in mind that the intended audience are the instructors and the TAs. Thus, for example, you should not spend two pages explaining what linear programming is.

Your milestone should include the full names of all your team members and state the full title of your project.

> **Note:** We will expect your final writeup to be on the same topic as your milestone.

Contributions Please include a section that describes what each team member worked on and contributed to the project. This is to make sure team members are carrying a fair share of the work for projects. If you have any concerns working with one of your project teammates, please create a private Ed post.

Grading The milestone is mostly intended to get feedback from the professor and your peers to make sure you're making reasonable progress. As long as your milestone follows the instructions above and you seem to have tested any assumptions which might prevent your team from completing the project, you should do well on the milestone.

Format Your milestone should be at most 3 pages, excluding references. Similar to the proposal, it should include:

- **Motivation:** What problem are you tackling, and what's the setting you're considering?
- **Method:** What OR techniques have you tried and why?

- **Preliminary experiments:** Describe the computational studies that you've run, the outcomes, and any error analysis that you've done. You should have tried at least one baseline.
- **Next steps:** Given your preliminary results, what are the next steps that you're considering?

Final Writeup

We know that most students work very hard on the final projects, and so we are extremely careful to give each writeup ample attention, and read and try very hard to understand everything you describe in it. After the class, we will also post all the final writeups online so that you can read about each other's work.

Format Final project writeups can be at most 5 pages long (including appendices and figures). We will allow for extra pages containing only references. If you did this work in collaboration with someone else, or if someone else (such as another professor) had advised you on this work, your write-up must fully acknowledge their contributions. For shared projects, we also require that you submit the final report from the class you're sharing the project with.

Contributions Same as for the Milestone submission.

Code Please include a link to a Github repository file with the code for your final project including all the details for the project to be executable without any additional files.

Grading The final report will be judged based off of the clarity of the report, the relevance of the project to topics taught in this course, the novelty of the problem, and the technical quality and significance of the work.

After the Course

After the course, if you want to submit your work to an operations research conference, the INFORMS and EURO deadlines are good places to start (INFORMS, EURO). Of course, depending on the topic of your project, other conferences may also be more appropriate.

Project FAQs

1. **What are the deliverables as part of the term project?**
   The project has four deliverables:

   a. Proposal

   b. Milestone

   c. Final report
      Please see the moodle site for due dates and deadlines.

2. **Should the final project use only methods taught in the class?**
   No, we don't restrict you to only use methods/topics/problems taught in class. That said, you can always consult a TA if you are unsure about any method or problem statement.

3. **Is it okay to use a dataset or problem instance that is not public?**
   We don't mind you using a dataset or instance that is not public, as long as you have the required permissions to use it. We don't require you to share the dataset either as long as you can accurately describe it in the Final Report. However, a toy model full implementation needs to be provided in GitHub.

4. **Is it okay to combine the term project with that of another class?**
   In general it is possible to combine your project for this course and another class, but with the following caveats:

   - You should make sure that you follow all the guidelines and requirements for this project (in addition to the requirements of the other class). So, if you'd like to combine your project with a class X but class X's policies don't allow for it, you cannot do it.
   - You cannot turn in an identical project for both classes, but you can share common infrastructure/code base/datasets across the two classes.
   - In your milestone and final report, clearly indicate which part of the project is done for this course and which part is done for a class other than this one. For shared projects, we also require that you submit the final report from the class you're sharing the project with.

5. **What are acceptable team sizes and how does grading differ as a function of the team size?**
   We recommend teams of 2-3 students. The team size will be taken under consideration when evaluating the scope of the project in breadth and depth, meaning that a three-person team is expected to accomplish more than a one-person team would.

   The reason we encourage students to form teams of 2-3 is that, in our experience, this size usually fits best the expectations for the projects. In particular, we expect the team to submit a completed project, so keep in mind that all projects require you to spend a decent minimum effort towards gathering data, and setting up the infrastructure to reach some form of result.

6. **What fraction of the final grade is the project?**
   The term project is 25% of the final grade.

7. **What is the late day policy for group project?**
   Students can use late days for both the proposal and milestone (though note that these late days apply to each member of the group). Also note that while late days are allowed, we do not recommend using them since the assignments are more time consuming than writing up the proposal and milestone (and worth more points).

8. **Can we use optimization libraries such as CPLEX, Gurobi, or are we expected to implement them from scratch?**
   You can use any library for the project.

9. **Is it ok to use a public repository for version control?**
   We recommend that students keep their repo private while working on the project. After the class ends, we understand that many students may want to have their work be public, so that they can point to it for interviews, outside advisors, etc, which is acceptable.

10. **What if two teams end up working on the same project?**
    It is okay if two teams end up working on the same project as long as they don't coordinate to do so, in order to not be biased in the way they tackle the problem. Alternatively the teams can coordinate to make sure they work on different problems.

11. **Will we be provided any cloud compute resource credit?**
    We are looking into getting cloud credit for the projects. We will announce here once this is finalized.

12. **Are we required to use a specific programming language for the project?**
    Yes, exclusively Python.