Decision-making in Transport and Logistics (1CM110)

ASSIGNMENT 2

Deadline: Jan 20th, 23:59

Dr. Layla Martin, Dr. Rolf van Lieshout, Dawei Chen

1. READ the **entire** assignment before you start. Get an idea of what you need to do and assess how much time you will need.
2. Do not split the work within you group alongside tasks or questions (e.g., person A works only on the report, person B only on the code, or person A works on question 1, and person B on question 2), but collaborate on all questions and all tasks.
3. It is **not** allowed to collaborate with people outside your group. If we observe this, all involved groups will be fined.
4. Copying code or text from others is considered plagiarism.
5. Test your implementations on small instances you constructed yourself, and verify that the results are correct. It is often good practice to write a little bit of verification code to verify the correctness of your solutions (don't just rely on your models).
6. If you solve your model on a (more or less) recent computer, and it is taking more than 30 minutes, something is wrong with your model.
7. Whenever you are using Gurobi, do NOT disable its output: you want to be able to see what's going on when the solver is solving your models.
8. All your Gurobi models must be linear models. Note that Gurobi does not throw an error if your model is non-linear, so you must check this yourself!
9. Code must be well-documented and easily readable. We will deduct points if we do not understand what the code does.
10. Submitted code must run immediately and provide all output as specified in the assignment, also on other computers. Make sure to clearly document (e.g. using a readme.txt) which libraries to install and ensure that your submission contains all necessary files.

11. Oftentimes, writing data to file during your implementation can help with later steps. Then, you only need to load the data instead of recomputing all steps.

12. This assignment is deliberately containing some weak specifications. You can use the Canvas discussions as well as the Q&A sessions to obtain clarity.

Goals of this assignment:

1. Enhance your understanding of the course material through hands-on experience.

2. Increase your experience with Python, and make you familiar with the implementation of small models in the commercial solver "Gurobi". Understand and interpret the Gurobi output.

3. Practice how to implement advanced algorithms in Python, using the example of several game theoretical algorithms.

4. Presenting actionable insights from sensitivity analyses.

Introduction

Last year, Uni Books B.V. suddenly found itself in competition with Book Import SE. Both companies supply multiple universities in the Netherlands. As a consequence, Uni Books B.V. aims at increasing their service level, lowering their costs, and at sustaining their customer base. Similar to Uni Books B.V., they also collect books from the printers, ship them via "cross-docks" (set C), and from there to universities. You can find supply, capacities, and demand in the file `demanddata.xslx`[1].

1 Reducing Transportation Cost and Fairly Allocating Them

Uni Books B.V.'s first idea is to reduce the cost they charge for transportation to their customers, i.e., the universities. Each of the universities shall be allocated a fair share of the fixed transportation cost, rather than full costs.

1.1. Preparation Steps (10 points)

Below, you find the mathematical model Uni Books B.V. currently uses to calculate transportation costs per university. There, P refers to all printers, W refers to all warehouses (each with a capacity of $\text{cap}(w) \forall p \in P$), U refers to all universities, and K refers to all types of books. The variable transportation costs c_{ij} between locations are given in `variablecosts.xlsx`, and the fixed transportation costs f_{ij} between locations are given in `fixedcosts.xlsx`.

[1] This file directly extends upon the demand data in the first assignment

For each university $u \in U$, Uni Books B.V. currently calculates the total costs using

$$\min \sum_{p \in P} \sum_{w \in W} \left(\sum_{k \in K} \left(c_{pw} x_{pw}^k \right) + f_{pw} y_{pw} \right) + \sum_{w \in W} \left(\sum_{k \in K} \left(c_{wu} x_{wu}^k \right) + f_{wu} y_{wu} \right) \quad (1)$$

$$\sum_{e \in \delta^+(i)} x_e^k - \sum_{e \in \delta^-(i)} x_e^k \leq o_i^k \qquad \forall i \in P, k \in K \quad (2)$$

$$\sum_{e \in \delta^+(u)} x_e^k - \sum_{e \in \delta^-(u)} x_e^k = -d_u^k \qquad \forall k \in K \quad (3)$$

$$\sum_{e \in \delta^+(i)} x_e^k - \sum_{e \in \delta^-(i)} x_e^k = 0 \qquad \forall k \in K, i \in W \quad (4)$$

$$\sum_{k \in K} x_{ij}^k \leq \min\{ \sum_{k \in K} o_i^k, \mathrm{cap}\,(j) \} y_{ij} \qquad \forall i \in P, j \in W \quad (5)$$

$$\sum_{k \in K} x_{iu}^k \leq \min\{ \mathrm{cap}\,(i), \sum_{k \in K} d_u^k \} y_{iu} \qquad \forall i \in W \quad (6)$$

$$x_{ij}^k \geq 0 \qquad \forall i, j \in P \cup W \cup \{u\}, k \in K \quad (7)$$

$$y_{ij} \in \{0, 1\} \qquad \forall i, j \in P \cup W \cup \{u\}, k \in K \quad (8)$$

(Hint: In this model, we only consider one university $u \in U$, and this is part of the input.)

(a) (1 point) What kind of model is this?

(b) (4 points) Define all sets, parameters, and variables that are not defined in the description.

(c) (1 point) Why does the model not consider arc capacities? What does this mean in a practical example?

(d) (4 points) Adapt the model to handle calculating costs for multiple universities simultaneously, and implement the adapted model in Python and Gurobi.

1.2. Cost Allocation (32 points)

To reduce transportation costs they charge to their customers, Uni Books B.V. now wants to allocate a fair share of their total costs to each customer.

(a) (2 points) How much do the transportation cost per customer and the total transportation cost deviate?

(b) (1 point) Which algorithm can you use allocate costs to the universities?

(c) (15 points) Fairly distribute the cost savings across all universities. What do the universities now pay, and how much do they save each? Implement the algorithm in Python, using Gurobi and the model you implemented before for calculating intermediate steps. Provide intermediate outputs with your code.

(d) (5 points) On a per-university level, does the source of the books and the paths books take change? Support your answer with visualizations.

(e) (5 points) Run your cost allocation algorithm also with the network structure of Book Import SE. For which of the customers is Uni Books B.V. cheaper, and for which of the customers is Book Import SE cheaper?

(f) (4 points) If each of the two companies only serves the customers for which they are the cheaper option, what price would they have to charge each customer to remain financially sustainable?

2 Collaborating and Competing

In the long run, competing on prices is not a viable approach for both companies. Instead, they leave the choice of the book shop to the final customer, i.e., the faculty and students of the universities. We henceforth assume that if both Uni Books B.V. and Book Import SE go to a customer location, the demand is split 50/50 between them, and so is the revenue of 6€ per book. It is possible that one of the companies only offers book 1, while the other offers all types of books. In this case, the demand for book 1 would be split equally, while the revenue of the other books only goes to the other company.

2.1. Preparatory Steps (10 points)

(a) (2 points) We assume that demand is split 50/50, if faculty and students (randomly) self-select whether to order via Uni Books B.V. or via Book Import SE. Why is this assumption reasonable? When would it not be reasonable?

(b) (4 points) Compute the joint profits if both companies collaborate, and jointly offer their service.

(c) (4 points) How much of the joint profit would go to Uni Books B.V., assuming profits are allocated in a fair way?

2.2. Competition (25 points)

While both companies would be better off by collaborating, strategic management of Uni Books B.V. decided to continue with their independent operations.

(a) (3 points) Explain why strategic management might prefer to continue competing with Book Import SE, rather than collaborating. Provide three reasons closely related to the book distribution industry.

(b) (15 points) Implement an iterated best-response algorithm as discussed in class using Python and Gurobi. How much profit can each of the two companies gain by considering competition (over ignoring it), and by collaborating (over competing)?

(c) (2 points) How much does Uni Books B.V. loose by competing with Book Import SE, rather than collaborating?

(d) (5 points) How do your answers to (b) and (c) change depending on the starting point of the best-response scheme?

2.3. Sensitivity Analysis (11 points)

Your analysis almost convinced strategic management to collaborate with Book Import SE. However, they have a few remaining questions.

(a) (3 points) Does the competitive position of Uni Books B.V. improve if they closed down one of the warehouses? Assume that closing a warehouse reduces the costs by 200.

(b) (5 points) Does the competitive position of Uni Books change if they adapt their delivery fees? Assume that if both companies serve a university, Uni Books B.V. only charges a fraction z of the original price (e.g., 80%, so 4.80), resulting in a market share of $0.5 + (1 - z)/2$ (e.g., 60% of the entire demand if $z = 80\%$). Book Import SE does not adapt delivery fees.
(Hint: you do not need to optimize over the price, but treat this as an input parameter to a sensitivity analysis)

(c) (3 points) Strategic management also suggests to reduce their offer. Instead of offering all types of books from all printers, they suggest to optimize which books to sell. How does this affect the competitive position of Uni Books B.V.?

3 Route Choice

Currently, Uni Books B.V.'s trucks from the warehouse W2 always arrive very late at university U3, since both highways the drivers can take are highly congested. Let $x = 6000$ be the number of vehicles traveling from W2 to U3 every hour, and x_1 be the number of vehicles using the first road, and x_2 the number of vehicles using the second road. The travel time along the first road is $75 + \frac{x_1}{120}$, while the travel time along the second road is $60 + \frac{x_2}{180}$.

(Hint: you can answer this entire block of questions using, for example, a sheet of paper, Excel, or similar. Python and Gurobi can also solve this problem, but might not be necessary)

3.1. Basic Analysis (4 points)

(a) (4 points) How long does it currently take to travel from U3 to W2, assuming that all vehicles independently select their path?

3.2. Expanding the Road Network (8 points)

The state considers multiple options to reduce congestion and thus travel times:

- expanding the throughput of the first road by adding more lanes, with a travel time of $75 + \frac{x_1}{60 \cdot n}$ with n the number of lanes
- incentivizing vehicles to travel elsewhere, resulting in a decreasing x
- increasing the maximum speed on the first road, reducing the travel time to $t + \frac{x_1}{120}$

They asked multiple transport companies, including Uni Books B.V., which option they would prefer, and why.

(a) (2 points) Calculate how the travel time reacts to additional lanes on the first road.

(b) (1 point) Explain why the travel time does not decrease linearly when adding more lanes.

(c) (2 points) Calculate how the travel time reacts to a reduced number of vehicles.

(d) (2 points) Calculate how the travel time reacts to an increased speed on the first road.

(e) (1 point) Which of the options should Uni Books B.V. suggest to the state?

4 Workload reporting

4.1. **Workload reporting** (1 point)

Please truthfully report

(a) who did which work

(b) how much work was it in total (in detail)

(this is a bonus point)

Reporting

Hand in the assignment via Canvas. Upload a **.pdf** file containing the complete report with answers to all questions (**NO MS Word documents**). Furthermore, hand in a .zip file containing your full Python code as well as Excel files you hand in. Make sure that your report (PDF) is both inside the zip archive and uploaded separately! Include assignment number, group number, and your last names in the filenames, e.g., `Assignment2_Group03_Martin_vanLieshout_Chen.pdf` and `Assignment2_Group03_Martin_vanLieshout_Chen.zip`. All your Python scripts (not Python notebooks!) must be executable irrespective of your computer and must produce clear output, and the file name must start with the exercise number, e.g., `2.2b_IBR.py`. The front page of your report must state your group number, and the names and student IDs of every group member. The final report cannot exceed 8 pages (including pictures etc., but excluding the front page and workload reporting). Deviation from these requirements can reduce your final grade by 1 point (out of 10).