OPTIMIZATION MODELING PROJECTS

Sonja Wogrin
Andrés Ramos
Sara Lumbreras
Félix Fernández
Miguel Álvarez
Ramón Rodrigáñez
Juan Javier Domínguez

September 2020

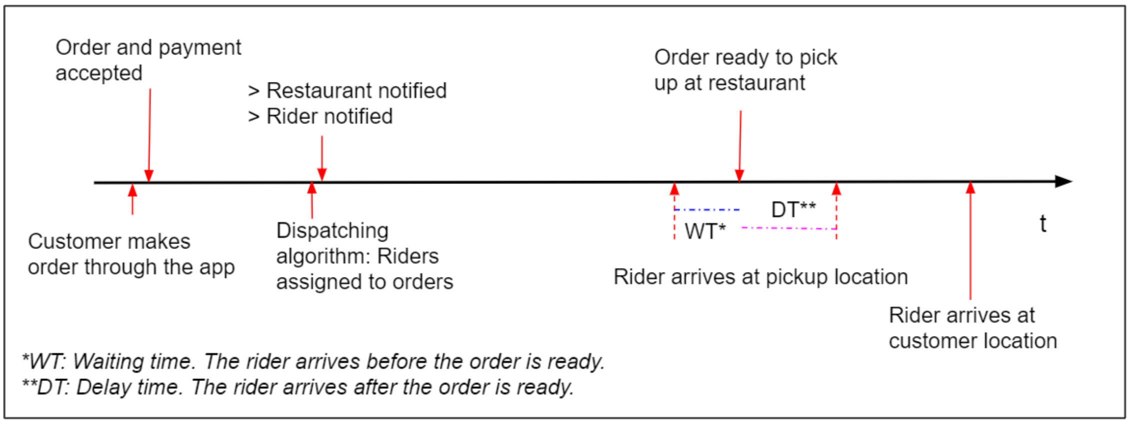
TABLE OF CONTENTS:

I. EasyEat

EASYEAT is a delivery company committed to delivering food from any restaurant in less than one hour. To achieve this, they rely on a mobile app that customers use to make their orders, a fleet of delivery men/women (which we will refer to as “rider”), and their crown jewel: the dispatching algorithm, an optimization algorithm responsible for assigning orders to riders.

The dispatching algorithm is run every minute. At every iteration (every time it is run), the algorithm takes all idle riders and the orders received during the last minute, together with the orders received in previous iterations that couldn’t be assigned for not having enough riders, and makes the optimal assignment.

EASYEAT has hired you to build an improved version of the dispatching algorithm. After an exhaustive analysis of the whole delivery process, they have come up with the following timeline describing the process:



In addition, they identified the following criteria that all assignments must fulfil:

1. At every iteration, one rider can only be assigned to one order.
2. At some iterations, there will be more riders than orders, and vice versa. It will always be compulsory to make as many assignments as possible.
3. At any point in time, the rider with the highest number of delivered orders cannot double the number of delivered orders of the rider with the lowest number plus one (highest number cannot exceed 2*[lowest number + 1]).
4. If an assignment is made such that the rider arrives at the restaurant 10 minutes after the order is ready, then, at least, for half of the other assignments, riders must arrive before their order is ready.

COMILLAS UNIVERSIDAD PONTIFICIA ICAI ICADE CIHS

5. Since Easyeat pays riders for waiting at the restaurant, and since we don't want food to get cold, the new objective function will minimize the following costs:
    a. Distance to pick up location (restaurant): 10€/km
    b. Delay at arrival at pickup location: 1€/minute of delay
    c. Waiting time at pickup location: 0.3€/minute waiting

In order to test your model, EASYEAT has provided you with the following data consisting of a set of orders and riders to be assigned at one iteration:

<table>
<tr><th colspan="9">TIME AND DISTANCE TO PICKUP (minutes | km)</th></tr>
<tr><th>Riders/Orders</th><th>O1</th><th>O2</th><th>O3</th><th>O4</th><th>O5</th><th>O6</th><th>O7</th><th>O8</th></tr>
<tr><td>R1</td><td>4.2 | 0.7</td><td>5.5 | 1.5</td><td>10 | 4.2</td><td>5.3 | 2</td><td>8 | 3</td><td>8.9 | 3.5</td><td>7.9 | 2.9</td><td>6.4 | 2</td></tr>
<tr><td>R2</td><td>8.9 | 4.7</td><td>6 | 3</td><td>5 | 2.4</td><td>6.3 | 3.2</td><td>7.7 | 1.7</td><td>3.9 | 0.9</td><td>3.7 | 1</td><td>5.4 | 2.6</td></tr>
<tr><td>R3</td><td>6.5 | 2.7</td><td>5.1 | 1.9</td><td>4.5 | 1.5</td><td>3.4 | 0.8</td><td>4.8 | 1.7</td><td>3.5 | 0.9</td><td>3.6 | 1</td><td>5.5 | 2.1</td></tr>
<tr><td>R4</td><td>8.6 | 2.8</td><td>7.1 | 1.9</td><td>6.5 | 1.5</td><td>5.5 | 0.9</td><td>6.9 | 1.7</td><td>5.4 | 0.8</td><td>5.5 | 0.9</td><td>7.5 | 2.1</td></tr>
<tr><td>R5</td><td>5.7 | 2.8</td><td>5.6 | 2.8</td><td>3.6 | 1.6</td><td>3 | 1.2</td><td>2.1 | 0.7</td><td>3.8 | 1.7</td><td>4.6 | 2.2</td><td>6.3 | 3.2</td></tr>
<tr><td>R6</td><td>5.9 | 2.3</td><td>2.6 | 4</td><td>7.8 | 3.5</td><td>5.2 | 1.9</td><td>7.5 | 3.3</td><td>6.2| 2.5</td><td>4.7 | 1.6</td><td>2.5 | 0.3</td></tr>
<tr><td>R7</td><td>26.6 | 4</td><td>25.6 | 3.4</td><td>20.7 | 0.4</td><td>23.5 | 2.1</td><td>23.1 | 1.9</td><td>22 | 1.2</td><td>23.6 | 2.2</td><td>26 | 3.6</td></tr>
<tr><td>R8</td><td>5.6 | 3.4</td><td>6.7 | 4</td><td>4.6 | 2.8</td><td>4 | 2.4</td><td>1.8 | 1.1</td><td>5.3 | 3.2</td><td>6.1 | 3.7</td><td>7.4 | 4.4</td></tr>
</table>

REMAINING ORDER PREPARATION TIME (minutes) (Time until orders are ready for pickup)							
O1	**O2**	**O3**	**O4**	**O5**	**O6**	**O7**	**O8**
7	10	2	15	5	20	5	15

NUMBER OF ORDERS DELIVERED SO FAR BY EACH RIDER							
R1	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**
5	5	7	4	6	8	3	4

Build a mixed-integer linear programming model to determine the optimal assignment given the criteria provided above. Test your model for the following two scenarios:

1. First, consider all orders and the first six riders.
2. Second, consider the first six orders and all riders.

II. Nova Talent

Nova Talent is the global top talent community which connects high potential individuals with the best companies. They are planning to host an event with 4 companies (Big Tech, Consulting, Banking and Startup) and 12 students coming from different universities and backgrounds.

The event is such that each company will organize 2 small workshops with 3 students each, one after the other, meeting in total 6 out of the 12 students. Attendees will thus meet 2 out of the 4 companies, one in each of the 2 workshops.

We have the following information about the universities, preferences and backgrounds of students attending the event:

Student	**University**	**Preferred Company 1**	**Preferred Company 2**	**Preferred Company 3**	**Preferred Company 4**	**Type**
Student 1	Uni 1	Big Tech	Consulting	Banking	Startup	Engineering
Student 2	Uni 1	Consulting	Banking	Big Tech	Startup	Engineering
Student 3	Uni 1	Consulting	Startup	Banking	Big Tech	Engineering
Student 4	Uni 2	Big Tech	Startup	Consulting	Banking	Marketing
Student 5	Uni 2	Big Tech	Banking	Startup	Consulting	Marketing
Student 6	Uni 2	Banking	Big Tech	Consulting	Startup	Marketing
Student 7	Uni 3	Banking	Consulting	Big Tech	Startup	Business
Student 8	Uni 3	Consulting	Banking	Startup	Big Tech	Business
Student 9	Uni 3	Startup	Banking	Consulting	Big Tech	Business
Student 10	Uni 4	Startup	Consulting	Big Tech	Banking	Business
Student 11	Uni 4	Consulting	Big Tech	Banking	Startup	Engineering
Student 12	Uni 4	Big Tech	Banking	Startup	Consulting	Marketing

We have also the following information about the preferred profiles for each company:

Companies	**Big Tech**	**Consulting**	**Banking**	**Startup**
Preferred Profile 1	Engineering	Business	Business	Marketing
Preferred Profile 2	Marketing	Engineering	Marketing	Engineering
Preferred Profile 3	Business	Marketing	Engineering	Business

Part 0 - Matching

We want to first build a "matching" table which will determine the global "satisfaction" that will generate the meeting of a student and a company. The satisfaction will be a score from 0 to 100, with up to 50 points of the score coming from the student preferences and up to 50 points from the company preferences for profiles. Build the satisfaction matrix from the data above, considering that:

- For student preferences: first preference will score 50, second 30, third 20, and fourth 0
- For company preferences: first preference will score 50, second 25, and third 0

Tip: no optimization is needed at this stage (since this is information that is known a priori; this is just data), just Excel / GSheets for example.

Part 1 - Optimize event satisfaction

Using the satisfaction table of part 0, find the assignment of students to workshops that maximizes total satisfaction considering that:

- In both workshops, each company will meet exactly 3 students in a different room
- Students should not repeat companies between workshops 1 and 2

Tip: formulate a binary integer program (BIP)

Part 2 - Ensure companies meet students from all universities

Modify the model above and find the new optimal solution if Nova promises all client companies that they will meet, at least, 1 student coming from each of the 4 universities during the whole event (considering the 2 workshops).

Part 3 - Ensure diversity at each room of the workshop

Modify the model above and find the new optimal solution if Nova also promises attending members that they will not be in the same room with people from their own university.

III. Loading train wagons with containers

We want to load containers in a train. Up to 3 containers can be loaded into each train wagon. The train composition may have up to 7 wagons, and we need to transport 15 containers. The weight in tons of each container is given in the following table.

Container	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
Weight [t]	5	7	5	6	8	8	7	5	7	8	4	4	5	3	2
Shipping	S1	S2	S3	S3	S1	S6	S6	S4	S5	S2	S3	S4	S5	S3	S6

Wagons can only transport up to 20 tons. We need to assign containers to wagons to minimize the maximum weight transported by any wagon.

To facilitate the container processing at the rail yard if several containers are loaded in the same wagon, they must belong to the same shipping company.

Due to some unknown external conditions, containers C10, C12 and C14 must necessarily go in different wagons.

Besides, if container C1 goes in wagon 1, then container C4 must go in wagon 6.

IV. Algorithms

The company Shannon Coding (SC) develops machine learning projects. They must set the developer teams to work in two similar projects for two different customers. One team of developers for each project.

SC has analysed both projects to determine which machine learning algorithm would be more suitable for each project. SC has concluded that any algorithm among random forest (RF), support vector machines (SVM), or neural networks (NN) would be fine. SC has now to decide what algorithm to use in each project (among those mentioned). Each project will use only one algorithm.

The developers available for the projects are shown in table 1, as well as their scores in each algorithm and their salary in euros per hour.

	Algorithm			Salary
Developer	RF	SVM	NN	€/hour
Anna	7	9	8	21
Barbara	7	6	8	18
Charles	8	8	8	21
David	9	8	7	21
Eleanor	8	8	9	22
Frederick	9	7	8	21
Gregor	8	6	7	18
Helen	6	8	7	18
Ian	7	8	8	20
Julia	9	8	9	23

Table 1. Developer information

The dedication of a developer is a value between 0 and 1 (dedication of 0.3 to a project means that the developer will work for that project the 30 % of their working time). A developer's dedication to a project is fixed for the duration of the project. If a developer participates in a project, they cannot do it with a dedication below 0.25 or above 0.75 to that project. Any developer can participate in both projects and they cannot have a total dedication above 1.

Each project has a maximum budget per hour: 80 €/h for project 1 and 60 €/h for project 2. The cost of a developer in a project is the result of multiplying their dedication to that project by the salary shown in table 1.

Each project has a fixed deadline and requires an exact dedication to meet the deadline. Project 1 requires an exact dedication equivalent to 4 full-time people and project 2 an exact dedication equivalent to 3 full-time people.

SC wants to maximize the sum of the total score of the projects. The contribution of a developer to the score of a project is the result of multiplying their dedication to that project by their score in the algorithm selected for that project. The total score of a project is the sum of the contribution of all developers participating in that project.

a) Formulate a mixed integer linear program to decide the dedication of each developer to each project and the algorithm selected for each project.
b) Modify the model from section a) to consider that the customer of project 1 demands that one developer (and only one) in their project must participate with full-time dedication. The rest of the developers will participate with a dedication between 0.25 and 0.75, as in section a). Compare these results with those from section a).

V. The vineyard in Ojacastro

We are the owners of a large vineyard in Ojacastro, La Rioja, and are considering installing new watering systems in some of the portions that compose the vineyard. We are deciding to install either 2 or 3 systems over 2 or 3 of the portions, which are 10 ha each. The total area is divided into several square portions of land where a new watering system can be installed. Each plot has an average increase in the annual production in tonnes that depends on the watering system being installed. The annual amortization of the water system in € if it is installed (number in the middle) and a cost of maintenance in € (number at the bottom) are also shown in the following table. These numbers have been estimated by our chief oenologist. The forecasted grape price for our variety of grapes is 0,85 € per kg.

Portions	**a**	**b**	**c**	**d**	**e**	**f**
X	30 t 3000 € 1000 €	40 t 4000 € 1200€	25 t 2500 € 900 €	28 t 2300 € 1100 €	35 t 3200 € 1200 €	26 t 3300 € 1300 €
Y	26 t 3200 € 1200 €	21 t 2700 € 800 €	27 t 4000 € 1400 €	40 t 5000 € 1500 €	35 t 2000 € 1200 €	40 t 4500 € 1500 €
Z	25 t 2800 € 1200 €	18 t 2000 € 700 €	20 t 2000 € 800 €	25 t 2800 € 1000 €	30 t 3000 € 1100 €	25 t 2200 € 900 €

Design a Mixed-Integer Programming problem to select optimally the water system installation to maximize the global profit of the vineyard. The total amortization should not exceed 3000 €. The total maintenance cost should be lower than one fifth of the total expected income.

Then, analyze the impact of the following investment conditions:

a) The portions where the system is installed should be contiguous (horizontal, vertical or diagonal).
b) If we select the portion (Y, a) then we need to select (X, a) and (X, b).

c) If the selected portion is in the column “a” the rest of portions should be selected in the same row.
d) If there is a selected portion in the row X”, if there are any other portions selected, they should belong to the same column.

VI. Shift Rostering

In a dairy production factory, the required number of employees to comply with the factory production schedule and regulatory measures are defined by the data science team. In a department, the employees all have the same role and skills and therefore are indistinguishable from each other for our purposes.

It is a task of the data science department to generate a weekly roster for this and other departments. A roster is formed by the shift assignments for each employee and day. The work is divided in three shifts, M, A and N, for morning, afternoon, and night, respectively. A free day is a non-working day and there are two types, a regular free day, represented by the symbol ".", and a free day after a night shift, represented by "-". A free day after a night shift is called a "nigh-off" shift. This difference between a "regular" free day and a nigh-off is relevant for the employees because they are considered differently when computing paid vacation days.

Shift requirements

Each week the number of employees needed by shift to keep the factory running are shown in Table 1.

Shift	Mo	Tu	We	Th	Fr	Sa	Su
M	3	2	2	2	2	3	3
A	2	2	2	2	2	3	2
N	1	1	1	1	2	3	2

Table 1: Employee requirement by day and shift.

Regulations

To comply with work and company regulations a roster must satisfy the following conditions:

a. Before a night shift there can only be a free day or a night-off.
b. Night-off shifts can only be assigned the day after a night shift.
c. An employee must have, at least, two or more consecutive free days during the week.
d. The number of employees working each day and shift must be greater or equal than the number specified in Table 1

Rotating system

Currently the rostering is done using a simple rotating system. Considering that there are fourteen employees in the department, the current roster is given in Table 2

Employee	Mo	Tu	We	Th	Fr	Sa	Su
1	M	M	M	M	M	.	.
2	.	M	M	M	M	M	.
3	M	.	.	.	.	M	M
4	.	.	.	.	.	M	M
5	A	A	A	A	A	.	.
6	.	A	A	A	A	A	.
7	A	.	.	.	.	A	A
8	.	.	.	.	.	A	A
9	N	-	N	-	N	-	M
10	.	N	-	N	-	.	N
11	M	.	.	.	.	N	-
12	.	.	.	.	.	N	-
13	.	.	.	.	.	N	-
14	.	.	.	.	N	-	N

Table 2: Rotating roster

Question 1

Write a mixed-integer linear programming model GAMS model that generates a rostering satisfying the previous work regulations and shift requirements. Show the roster is a similar way as it is shown in Table 2.

Question 2

Modify the optimisation model to find the minimum number of employees required to satisfy all the constraints considered in the previous question. Show the roster in a similar way as it is shown in Table 2.