BAN402

Decision Modelling in Business

Project 2

19.10.2022

Group number:
26

Group members:
35
64

Table of Contents

Part A Chessmon GO

Task 1 Identifying procedure

The Karlsen family wish to minimize the walking distance while collecting all 128 pieces together. They have interpreted the task at hand and concludes that each of them shall collect at least 15 pieces each, and at most 50 pieces each. The family consists of four persons.

Firstly, we would need to build a model that can split all the locations into 4 sets, while grouping them most effectively. Set partitioning model can do this job.

The second model would be a symmetric TSP model that minimizes travelling distance while giving the most optimal trip (which order to go through each location). This model would be run one time for each person.

Model 1: Set Partitioning Model

The set partitioning model can help us choose 4 subsets of locations that the persons shall visit. The subsets are based on an optimal allocation of locations, based on minimizing total distance between all locations. The model does not consider which order the locations should be visited in, but that is covered in model 2 TSP model. This model does the work based on the assumption that the family has available all the necessary data and abilities to set up the data correctly for the model to be run.

The general model notation is as follows:

Sets

I	Set of locations
J	Subsets of all location combinations between 25 and 50 locations.

Parameters

a_{ij}	Binary, 1 if location i is included in subset j
c_j	Total distance between all locations in subset j

Decision variable

x_j	Binary, 1 if subset j is used

Objective function

$\min z = \sum_{j \in J} c_j x_j$ — Minimize total distance for all used subsets

Constraints

$\sum_{J \in J} a_{ij}\, x_j = 1 \qquad i \in I$ — Each location must be included exactly once

$\sum_{j \in J} x_j = 4$ — Number of used sets must be equal to the number of people available

$x_j\,, a_{ij} \in \{0,1\} \qquad i \in I, j \in J$ — x_j & a_{ij} are binary variables

The parameter c_j consists of the sum of all distances the connects the locations in a subset. This way the model can make subsets of locations that is close to each other in the most optimal way.

Model 2: Travelling Salesman Problem (TSP)

The TSP model enable us to make the optimal trip that minimizes the distance travelled while visiting each location. The general model notation is described below:

Sets

K — Set of all links between each location

c_k — The distance of each link k, $\forall\ k \in K$

Decision variables:

$x_k = \begin{cases} 1 \\ 0 \end{cases}$ — 1 if link k is included in the TSP trip; 0 otherwise

Objective function:

$minimize\ z = \sum_{k \in K} c_k x_k$ — The objective is to minimize the total distance of each link that is included in the trip

Constraints:

$\sum_{k \in L_j} x_k = 2 \qquad \forall\ j \in N$ — All location needs two linkages, and therefore is all locations visited exactly once

$\sum_{k \in B(S)} x_k \leq |\mathrm{S}| - 1 \qquad \forall\ \mathrm{S}\ \subset \mathrm{N}\!: |\mathrm{S}| \geq 2$ — Prevents subsets and makes one trip through all locations.

$x_k \in \{0,1\}$ — x is a binary variable of 1 or 0

We believe that the symmetric version of the TSP model will do the job since there is no diversion in distance depending on which way you travel from one location to another.

Task 2 Jacob's claim

Jacob claims that he will travel less distance than all of the family members will travel together. Intuitively, one would think that Jacob would travel less distance than the family since he would not need to travel back to the hotel the same number of times as the family members. Even though he will have other distances to travel, from one subset of locations to another, he will never travel more than the family.

Part B Exam schedule

Task 1 Relatives

This task introduces the scenario where candidate c1 and professor p1 cannot be assigned to the same time slot where the candidate takes an exam. We would solve this by introducing the following constraint:

$$x_{et} + z_{pt} \leq 1 \qquad \forall \; t \in T, \qquad e \in E_{C^{(1)}}, \qquad p = 1$$

Task 2 Bad-luck sequences

First condition:

To consider the first condition, we introduce a new binary variable (1) that takes the value 1 if a candidate experiences a bad luck sequence. This requires one new constraint (2) that "activates" the variable. This variable is used in a new constraint (3) that makes sure the sum of bad luck sequences is less than or equal to two for each student.

New binary variable u:

$$u_{et} \begin{cases} 1 \; if \; candidate \; c \; encounters \; bad \; luck \; sequence \\ 0 \; if \; otherwise \end{cases} \qquad (1)$$

Variable u_{et} takes value 1 in day d if candidate c is assigned exam in day d and $d + 2$. If a candidate is assigned exams in both day d and d+2 the left side will be equal to 2, which requires u to take the value 1.

$$\sum_{e \in E_C(c)} \; \sum_{t \in T(d) \, \cup \, T^{d+2}} x_{et} \geq 2u_{et} \qquad (2)$$

Each candidate cannot experience more than two bad luck sequences:

$$\sum_{e \in E_{C(c)}} \sum_{t \in T} u_{et} \leq 2 \qquad (3)$$

The second condition is that at least 60% of the candidates have no bad-luck sequences. We can solve this by introducing a new variable r_{et} which takes 1 if a candidate experiences bad luck sequences 1 or two times.

$$r_{et} \begin{cases} 1 \ if \ candidate \ c \ encounters \ bad \ luck \ sequence \ one \ or \ two \ times \\ 0 \ if \ otherwise \end{cases}$$

Further we need three constraints more, where two is initiating the variable r and one for limiting the percentage amount of candidates having bad-luck sequences.

r_{et} takes the value 1 if a candidate experiences at least one bad luck sequence, zero otherwise

$$\sum_{e \in E_{C(c)}} \sum_{t \in T} u_{et} + 1 \geq 2r_{et} \tag{1}$$

$$\sum_{e \in E_{C(c)}} \sum_{t \in T} u_{et} \leq 2r_{et} \tag{2}$$

No more than 40% of candidates can have bad luck sequences.

$$\sum_{e \in E} \sum_{t \in T} r_{et} \leq \ 0.4 \cdot |C| \tag{3}$$

$|C|$ is the cardinality of C

(Number of candidates in C)

Task C PetroBAN

Task 1 Formulating and implementing the model

PetroBAN refines crude oils, blends products, and sells their products in different markets with an objective of maximizing profit. This problem can be formulated as follows:

Sets	
$i \in I$	Set of crude oils
$b \in B$	Set of components
$j \in J$	Set of CDUs
$m \in M$	Set of modes
$p \in P$	Set of products
$d \in D$	Set of depots
$k \in K$	Set of markets
$t \in T$	Set of days
Parameters	
R_{ibjm}	Amount of component b obtained from crude i in CDU j in mode m
CAP_{jm}	Processing capacity of crude oils in CDU j in mode m per day
N_{bp}	Required units of component b in product p
δ_{pkt}	Max demand for product p in market k on day t
S_p	Sales price per unit product p in all markets
S^{low}	Sales price per unit lowqc
C^{change}	Cost of changing modes between days
C_{jm}^{mode}	Cost of operating CDU j in mode m per day
C_{ijm}^{ref}	Cost of refining one unit crude oil i in CDU j in mode m
C_{it}^{crude}	Cost of purchasing one unit crude oil i on day t
C_p^{prod}	Cost of producing one unit of product p
C^{tra1}	Cost of transporting one unit of any component to blending
C_d^{tra2}	Cost of transporting one unit of any product to depot d
C_{dk}^{tra3}	Cost of transporting one unit of any product from depot d to market k
C^{invi}	Daily cost of storing one unit of any crude in the refining department
C^{invb}	Daily cost of storing one unit of any component in the refining department
C_d^{invp}	Daily cost of storing one unit of any product in depot d
I_{pd}^{zeroP}	Initial inventory of product p in depot d on day $t = 0$
I_{pd}^{finalP}	Final minimum inventory of product p at depot d on day $t = 12$
I_i^{zeroO}	Initial inventory of crude i in refinery
I_i^{finalO}	Final minimum inventory of crude I in refinery
I_b^{zeroC}	Initial inventory of component b in refinery
I_b^{finalC}	Final minimum inventory of component b in refinery
S^{low}	Sales price of lowqc component
Decision variables	
x_{it}	Amount of crude i to purchase and send to refinery on day t
w_{it}	Amount of crude i processed on day t
y_{ijmt}	Amount of crude i processed using CDU j in mode m on day t
z_{bt}	Amount of component b shipped to blending on day t
v_{pdt}	Amount of product p produced and sent to depot d on day t
u_{pdkt}	Amount of product p sold to market k from depot d on day t
IO_{it}	Amount of crude i in refinery storage on day t
IC_{bt}	Amount of component b in refinery storage on day t
IP_{pdt}	Amount of product p in storage at depot d on day t
G_{jmt}^{mode}	binary variable, 1 if CDU j uses mode m on day t, 0 otherwise
G_{jmt}^{change}	binary variable, 1 if CDU j changes mode m on day t from day t-1

Objective function
Maximize profit:

$$\sum_{p\in P}\sum_{d\in D}\sum_{k\in K}\sum_{t\in T:t>0\ and\ t\leq 11} S_p u_{pdkt} + \sum_{i\in I}\sum_{b\in B:b="lowqc"}\sum_{j\in J}\sum_{m\in M}\sum_{t\in T:t>0} S^{low} R_{ibjm} y_{ijmt} - \sum_{i\in I}\sum_{t\in T:t>0} C_{it}^{crude} x_{it} -$$
$$\sum_{p\in P}\sum_{d\in D}\sum_{t\in T:t>0} C_p^{prod} v_{pdt} - \sum_{b\in B}\sum_{t\in T:t>0} C^{tra1} z_{bt} - \sum_{p\in P}\sum_{d\in D}\sum_{t\in T:t>0} C_d^{tra2} v_{pdt} -$$
$$\sum_{p\in P}\sum_{d\in D}\sum_{k\in K}\sum_{t\in T:t>0} C_{dk}^{tra3} u_{pdkt} - \sum_{i\in I}\sum_{t\in T:t>0} C^{invi} IO_{it} - \sum_{b\in B}\sum_{t\in T:t>0} C^{invb} IC_{bt} - \sum_{p\in P}\sum_{d\in D}\sum_{t\in T:t>0} C_d^{invp} IP_{pdt} -$$
$$\sum_{j\in J}\sum_{m\in M}\sum_{t\in T:t>0} C^{change} G_{jmt}^{change} - \sum_{j\in J}\sum_{m\in M}\sum_{t\in T:t>0} C_{jm}^{mode} G_{jmt}^{mode} - \sum_{i\in I}\sum_{j\in J}\sum_{m\in M}\sum_{t\in T:t>0} C_{ijm}^{ref} y_{ijmt}$$

Constraints
Balance of crude oils at refinery

$$IO_{it} = IO_{it-1} + x_{it} - \sum_{j\in J}\sum_{m\in M} y_{ijmt} \qquad \forall\, i\in I, t\in T: t>0 \tag{1}$$

Max processing capacity of total crude oil that can be processed each day

$$\sum_{i\in I} y_{ijmt} \leq CAP_{jm} \qquad \forall\, j\in J, m\in M, t\in T: t>0 \tag{2}$$

Balance of components at the distillation department

$$IC_{bt} = IC_{bt-1} - z_{bt} + \sum_{i\in I}\sum_{j\in J}\sum_{m\in M} R_{ibjm} y_{ijmt} \qquad \forall\, b\in B, t\in T: t>0\ and\ b = "lowqc" \tag{3}$$

Quantity sent to blending is according to required recipes (no storage in blending)

$$z_{bt-1} = \sum_{p\in P}\sum_{d\in D} N_{bp} v_{pdt} \qquad \forall\, b\in B, t\in T: t>0 \tag{4}$$

Balance of products at depots

$$IP_{pdt} = IP_{pdt-1} + v_{pdt-1} - \sum_{k\in K} u_{pdkt} \qquad \forall\, p\in P, d\in D, t\in T: t>0 \tag{5}$$

Quantity sent to markets cannot be greater than demand for each product in each market

$$\sum_{d\in D} v_{pdkt-1} \leq \delta_{pkt} \qquad \forall\, p\in P, k\in K, t\in T: t>0 \tag{6}$$

Mode variable logic, takes the value 1 if crude j is processed at CDU 1 using mode m at time t

$$y_{ijmt} \leq CAP_{jm} G_{jmt}^{mode} \qquad \forall\, i\in I, j\in J, m\in M, t\in T: t>0 \tag{7}$$

Mode variable logic, a CDU can only use one mode each day

$$\sum_{m\in M} G_{jmt}^{mode} = 1 \qquad \forall j\ in\ J, t\in T \tag{8}$$

Change variable logic, takes the value 1 if the CDU j is changed to mode m in time t

$$-G_{jmt-1}^{mode} + G_{jmt}^{mode} \leq G_{jmt}^{change} \qquad \forall\, j\in J, m\in M, t\in T: t>0 \tag{9}$$

Initial and final values

$$\mathrm{IO}_{i,t} = I_i^{zeroO} \qquad \forall\, \mathrm{i} \in \mathrm{I}, t\in T: t=0 \tag{10}$$
$$IC_{b,t} = I_b^{zeroC} \qquad \forall\, b\in B, t\in T: t=0 \tag{11}$$
$$IP_{pdt} = I_{pd}^{zeroP} \qquad \forall\, p\in P, d\in D, t\in T: t=0 \tag{12}$$
$$z_{bt} = 0 \qquad \forall\, b\in B, t\in T: t=0 \tag{13}$$
$$v_{pdt} = 0 \qquad \forall\, p\in P, d\in D, t\in T: t=0 \tag{14}$$
$$u_{pdkt} = 0 \qquad \forall\, p\in P, d\in D, k\in K, t\in T: t=0 \tag{15}$$
$$IO_{it} = I_i^{finalO} \qquad \forall\, \mathrm{i} \in \mathrm{I}, t\in T: t=12 \tag{16}$$
$$IC_{bt} = I_b^{finalC} \qquad \forall\, b\in B, t\in T: t=12 \tag{17}$$
$$IP_{pdt} = I_{pd}^{finalP} \qquad \forall\, p\in P, d\in D, t\in T: t=12 \tag{18}$$
$$G_{jmt}^{mode} = 1 \qquad \forall\, j\in J, m\in M, t\in T: t=0\ and\ m = "shutdown" \tag{19}$$

Non-negativity condition and binary variables

$$x_{it}, y_{ijmt}, z_{bt}, v_{pdt}, u_{pdkt}, IO_{it}, IC_{bt}, IP_{pdt} \geq 0 \qquad \forall\, i\in I, j\in J, m\in M, p\in P, d\in D, k\in K, b\in B, t\in T \tag{20}$$
$$G_{jmt}^{mode}, G_{jmt}^{change} \in \{0,1\} \qquad \forall\, j\in J, m\in M, t\in T \tag{21}$$

Profit

By implementing the model in AMPL and running it with CPLEX solver, the resulting total profit is 5 200 630.

Modes

Figure 1 displays the mode plan for CDU 1 and CDU 2. CDU1 doesn't run low mode, while CDU 2 runs low mode at day 2 and day 8 through 10. CDU1 is running high mode day 1 through 8, while CDU 2 runs high mode day 1, day 3 through 7 and day 12. The remaining days are spent in shutdown.

	day												
	0	1	2	3	4	5	6	7	8	9	10	11	12
CDU 1	S	H	H	H	H	H	H	H	H	S	S	S	S
CDU 2	S	H	L	H	H	H	H	H	L	L	L	S	H

Shutdown Low High

Figure 1:CDU mode plan original

Capacity

Table 1 displays each CDUs refining plan. The bottom of each plan summarizes how much of the corresponding mode capacity is being used. Note that the exact numbers contain more decimals, the numbers were rounded in table 1.

	day												
CDU1	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
CrA	0	0	0	0	0	0	0	0	0	0	0	0	0
CrB	0	929,1	950	950	918,3	950	562,3	919,9	924,5	0	0	0	0
sum	0	929,1	950	950	918,3	950	562,3	919,9	924,5	0	0	0	0
cap	0	950	950	950	950	950	950	950	950	0	0	0	0
%		97,8 %	100 %	100 %	96,7 %	100 %	59,2 %	96,8 %	97,3 %				

	day												
CDU 2	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
CrA	0	315,9	1000	574,5	657,6	628,2	796,9	94,3	1000	1000	395	0	136,7
CrB	0	584,1	0	325,5	242,4	271,8	103,1	805,8	0	0	0	0	65
sum	0	900	1000	900	900	900	900	900	1000	1000	395	0	201,7
cap	0	900	1000	900	900	900	900	900	1000	1000	1000	0	900
%		100 %	100 %	100 %	100 %	100 %	100 %	100 %	100 %	100 %	39,5 %		22,4 %

Table 1: Each CDUs refining plan

Final inventories

Our model never chooses to store any crude oil in the refinery, resulting in a final inventory of crude oils equal to zero. The final inventory of components is a bit more interesting. There is a minimum final inventory of 80, which is the final inventory of both naphtha 1 and naphtha 2. Distil A and distil B is both above this minimum quantity at 1 589.74 and 1061.47, respectively. Since lowqc is sold immediately after its refining process, the inventory of this component is always zero.

The main reason for this excessive inventory of distil A and B is that the naphtha components are used the most in blending, while all components are obtained from the same refining process. In the process of obtaining enough naphtha 1 and 2 to meet the minimum inventory, extra distil components are obtained as a side product.

Looking at the final product inventory in both depots, we find that the quantities are exactly equal to the specified minimum inventory. This is in line with expectations since the objective is to maximize profit. There is no added profit in having higher final inventories.

Task 2 Analysing three scenarios

PetroBAN wishes to analyse three different scenarios regarding inflation in the crude oil prices. In order to run these scenarios, we introduce a new parameter $C^{inflation}$. This parameter will take the value 1+inflation. Thus, $C^{inflation}$ will be 1.1 in scenario 1, 1.3 in scenario 2 and 1.5 in scenario 3.

The parameter is implemented in the objective function, where the total cost of crude oil purchases is calculated:

$$\sum_{i \in I} \sum_{t \in T: t>0} C_{it}^{crude} x_{it} \rightarrow \sum_{i \in I} \sum_{t \in T: t>0} C_{it}^{crude} C^{inflation} x_{it}$$

In AMPL, a new .mod file is created for each scenario, containing the new parameter "Cinflation" with its corresponding value and the new sum of inflated crude oil expenses in the objective function.

Originally, we had an optimal profit of 5 200 630.

In scenario 1, the optimal profit is reduced to 5 126 080. Thus, the profit reduction in this scenario compared to the original is 74 550.

In scenario 2, the optimal profit is reduced to 4 978 100. Thus, the profit reduction in this scenario compared to the original is 322 530, or 247 980 compared to scenario 1.

In scenario 3, the optimal profit is reduced to 4 832 680. Thus, the profit reduction in this scenario compared to the original is 367 950, Scenario 2 is 45 420, or 293 400 compared to scenario 1.

In figure 2 we see the optimal CDU mode plan for scenario 1, which is the same as the original plan. In figure 3 and figure 4 we see that the CDU mode plans change for scenario 2 and 3.

day

	0	1	2	3	4	5	6	7	8	9	10	11	12
CDU 1	S	H	H	H	H	H	H	H	H	S	S	S	S
CDU 2	S	H	L	H	H	H	H	H	L	L	L	S	H

Shutdown Low High

Figure 4: CDU mode plan scenario 1

	0	1	2	3	4	5	6	7	8	9	10	11	12
CDU 1	S	H	H	H	H	H	H	H	H	H	S	S	S
CDU 2	S	H	H	H	H	H	H	H	H	L	L	L	L

Figure 3: CDU mode plan scenario 2

	0	1	2	3	4	5	6	7	8	9	10	11	12
CDU 1	S	H	H	H	H	H	H	H	H	H	S	S	S
CDU 2	S	H	H	H	H	H	H	H	H	H	H	S	L

Figure 2:CDU mode plan scenario 3

Part D Optidriver

Task 1 LP model

Optidriver aims to minimize their EV charging costs over a three-month period. Our general model notation is as follows:

Sets

$m \in M$	set of months
$d \in D$	set of days
$h \in H$	set of hours

Parameters

c_{mdh}	Cost pr hour of charging in month m, day d and hour h, in EUR/kWh
a_{mdh}	Available charging in month m, day d at hour h
$v_{m,d,h}$	Energy usage in month m, day d at hour h

Decision variables

X_{mdh}	Charging amount in month m, day d and hour h
b_{mdh}	State of charge in month m, day d and hour h

Objective function

Minimize cost:

$$\sum_{m \in M, d \in D, h \in h} c_{mdh} * x_{mdh}$$

Constraints

Maximum battery charge rate

$$x_{mdh} \leq a_{mdh} \qquad \forall\, m \in M, d \in D, h \in H \qquad (1)$$

Maximum battery capacity

$$b_{mdh} \leq 64 \qquad \forall\, m \in M, d \in D, h \in H \qquad (2)$$

Minimum battery charge

$$b_{mdh} \leq 12.8 \qquad \forall\, m \in M, d \in D, h \in H \qquad (3)$$

Balancing battery level between hours

$$b_{mdh} = b_{mdh-1} + x_{mdh} - v_{mdh} \qquad \forall\, m \in M, d \in D, h \in H : h > 1 \qquad (4)$$

Balancing battery level between days

$$b_{md,1} = b_{m,d-1,24} + x_{md1} - v_{md1} \qquad \forall\, m \in M, d \in D : d > 1 \qquad (5)$$

Balancing battery level between July and August

$$b_{\mathrm{Aug},1,1} = b_{Jul,31,24} + x_{\mathrm{Aug},1,1} - v_{\mathrm{Aug},1,1} \qquad (6)$$

Balancing battery level between August and September

$$b_{Sep,1,1} = b_{Aug,31,24} + x_{Sep,1,1} - v_{\mathrm{Aug},1,1} \qquad (7)$$

Initial state of charge

$$b_{jul,1,1} = 51.2 + x_{jul,1,1} \qquad (8)$$

Final state of charge

$$b_{sep,31,1} = 51.2 \qquad (9)$$

The linear programming model results is presented and analysed with some calculations in the excel file “Task D results and calcualtions.xlsx”. The results and calculations are also illustrated in the table below.

	Optimal solution	Average price incurred by the optimal solution
Minimizing problem	116,743	168,617
Maximizing problem	272,579	374,003
Diff in absolute terms	155,836	205,386
Diff in relative terms	233,5%	221,8%

Task 2 Mixed integer linear model

To record weekend trips, we need a mixed integer model. Changes to the previous model is summarized below:

New binary variables:

$o_{mdh} \in \{0,1\}$	"out" trip	1 if car leaves home in month m, day d, hour h, zero if otherwise
$r_{mdh} \in \{0,1\}$	Recreational activity	1 if car is used for recreational activities in month m, day d, hour h
$i_{mdh} \in \{0,1\}$	"in" trip	1 if car returns to home in month m, day d, hour h, zero if otherwise
$z_{imh} \in \{0,1\}$	Charge availability	1 if car can charge in month m, day d, hour h, zero if otherwise

New parameter:

$k = 2.75$	Weekend drainage	Amount of energy used on a single weekend trip

New constraints:

Logic 1: o_{mdh} takes value 1 if a weekend trip is initiated, zero if otherwise

$$r_{mdh} - r_{mdh+1} - 1 \geq -1 - o_{mdh} \qquad \forall\, m \in M, d \in D, h \in H: d = sundays\ and\ h < 24 \qquad (10)$$

Logic 2: i_{mdh} takes value 1 if a weekend trip is ending, zero if otherwise

$$r_{mdh} - r_{mdh-1} - 1 \geq \ -1 - i_{mdh} \qquad \forall\, m \in M, d \in D, h \in H: d = sundays\ and\ h > 1 \qquad (11)$$

Logic 3: r_{mdh} takes value 1 if a trip was initiated previous hour or ends in the following hour, zero if otherwise

$$r_{mdh} = o_{mdh-1} + i_{mdh+1} \qquad \forall\, m \in M, d \in D, h \in H: d = sundays\ and\ h > 1\ and\ h < 24 \qquad (12)$$

Logic 4: z_{mdh} takes value 1 if there is no weekend trip, zero if otherwise

$$z_{mdh} = 1 - o_{mdh} - r_{mdh} - i_{mdh} \qquad \forall\, m \in M, d \in D, h \in H: h > 1\ and\ h < 24 \qquad (13)$$

Updated constraints:

Balancing battery level between hours

$$b_{mdh} = b_{mdh-1} + x_{mdh} - v_{mdh} - o_{mdh} * k - i_{mdh} * k \qquad \forall\, m \in M, d \in D, h \in H: h > 1 \qquad (4)$$

Balancing battery level between days

$$b_{md,1} = b_{m,d-1,24} + x_{md1} - v_{md1} - 0_{mdh} * k - i_{mdh} * k \qquad \forall\, m \in M, d \in D, h \in H: h > 1 \qquad (5)$$

"Illegal" constraint:

Initiative: Optidriver initiates recreational activities every Sunday at 11:00

$$o_{mdh} = 1 \qquad \forall\, m \in M, d \in D, h \in H: d = sundays\ and\ h = 12 \qquad (14)$$

The new binary variables o , r , i records when Optidriver is using the car in the weekend. z allows us to deny charging when the car is being used. The parameter k is used in combination with the binary variables to reduce battery level when the car is being used in the weekends in the updated constraint (4).

We were unable to find a logical constraint that lets Optidriver optimize when to use the car in the allowed timeframe. Therefore, we introduced an "illegal" constraint (14) that initiates recreational trips every Sunday at 11:00. This results in an alteration to the given problem formulation, as the model is unable to "freely" choose when to charge during the recreational window 10:00 – 17:00.

These constraints result in an optimal solution of 138.804 EUR. The average charging price incurred by the optimal solution are 182.228 EUR / MWh. These calculations can be seen in file "Task D results and calculations".

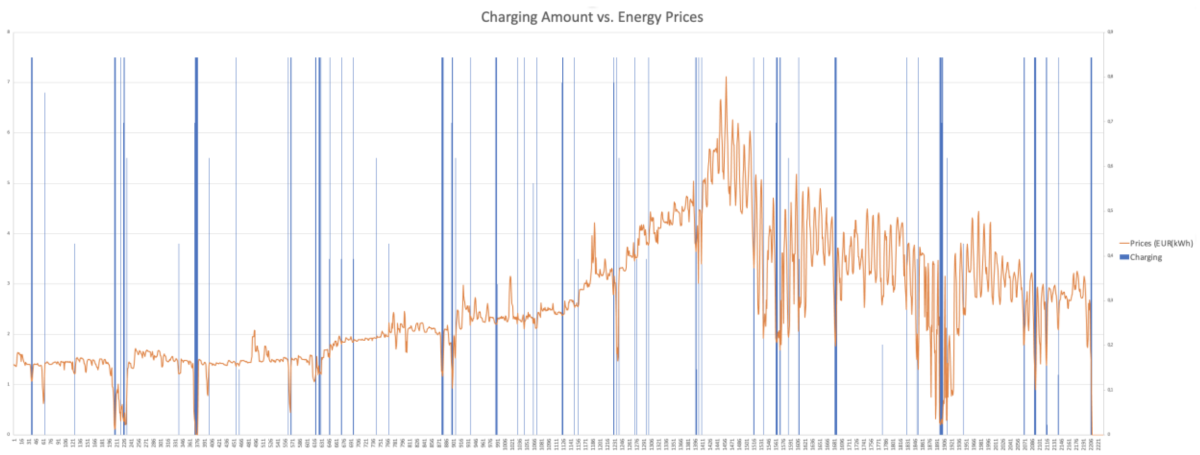

Figure 5: Charge vs. Price

Figure 5 plots charging amount each hour against the energy prices each hour. There is a substantial relationship between energy prices and charging amount. Whenever the energy prices is low, Optidriver charges the EVs battery more than when the prices are high.

Task 3: Home office

Task 4: Concerned about battery life