Assignment 7

March 11, 2024 1:41 PM

MIE 376 Assignment 7 Winter 2024

In this assignment, you model and solve problems for mathematical programming. You need to
(1) Formulate MP models.
(2) Implement the MP models using Python and Gurobi.

1. Submission Instructions

Submit a PDF file describing mathematical programming formulations and optimal solutions to the problem instance. Also, submit a program (a Python script or a Jupyter notebook) using Gurobi to solve the problem instance. In the MP formulation, clearly define decision variables and state the objective function and constraints. For the problem instance, report the values of the objective and decision variables.

2. IP

A car factory crafts three distinct varieties of cars: the nimble "Sprint", the versatile "Voyager", and the majestic "Titan". For each of these models, the requirements for steel and labor hours, along with the revenue and variable costs per unit, are outlined in the accompanying table.

Car Model	Sprint	Voyager	Titan
Steel per Unit (tons)	1.5	3	5
Labor Time per Unit (hours)	280	250	320
Revenue per Unit (k$)	11	14	16
Variable Costs per Unit (k$)	8	10	11

At the factory, the monthly availability of steel is 600t and monthly total labor hours are 60,000h.

Question 1: assuming an absence of fixed costs, develop a monthly production plan to maximize the profit of factory. Build the model formulation and solve the model with Gurobi.

Question 2: under certain limitations, if a certain model of car is to be produced, at least 70 units must be manufactured. How should the optimal production plan be changed? Build the model formulation and solve the model with Gurobi.

Question 3: based on **Q1**, we now know that once producing a certain model or a combination of models, an overall fixed cost will be incurred as the table shows.

Model to Produce	{S}	{V}	{T}	{S,V}	{S,T}	{V,T}	{S,V.T}
Overall Fixed Cost (k$)	180	240	240	300	420	400	450

For instance, the fixed cost for producing only the "Sprint" is $180k, while for producing just the "Voyager" it stands at $240k. If the decision is made to manufacture both the "Sprint" and "Voyager" but not the "Titan", the total fixed cost amounts to $300k. Build the model formulation and solve the model with Gurobi.

→ SOLUTION BELOW AND ON PYTHON - GUROBI

3. Dynamic Programming

Formulate and solve problem 7 from DP problem set. You can solve the problem either by hand or using python.

→ SOLUTION BELOW

Problem 1

Let st_i be the amount of steel required for 1 unit of car i — $i \in \{\text{sprint}, \text{voyager}, \text{titan}\}$ (s) (v) (t)

Let lb_i be the amount of labour required for 1 unit of car i

Let u_i be the number of units of car i produced. → u_s, u_v, u_t → decision variables

Let r_i be the revenue on selling for 1 unit of car i

Let v_i be the variable cost on production for 1 unit of car i

Objective Function to max profit

Objective: $\max \sum_i^3 r_i u_i - \sum_i^3 v_i u_i$ ⇒ $\max\ (11u_s + 14u_v + 16u_t) - (8u_s + 10u_v + 11u_t)$

$\sum_i^3 st_i u_i \le 600$ — s.t. $1.5u_s + 3u_v + 5u_t \le 600$

$\sum_i^3 lb_i u_i \le 60000$ — $280u_s + 250u_v + 320u_t \le 60000$

Constraints for steel and labour

Problem Formulation

Optimal Solution to Problem Instance

Objective value: in K\$ 864 = \$864000 — profit in \$

No. of Sprint cars produced (u_s) = 64

No. of voyager cars produced (u_v) = 168

No. of titan cars produced (u_t) = 0

cars of each type produced optimally.

Problem 2

Let st_i be the amount of steel required for 1 unit of car i — $i \in \{\text{sprint}, \text{voyager}, \text{titan}\}$ (s) (v) (t)

Let lb_i be the amount of labour required for 1 unit of car i

Let u_i be the number of units of car i produced. → u_s, u_v, u_t → decision variables

Let p_i be the binary variable $\{0,1\}$ to indicate if car i is produced or not → p_s, p_v, p_t

Let r_i be the revenue on selling for 1 unit of car i

Let v_i be the variable cost on production

Let r_i be the revenue on selling
for 1 unit of car i

Let v_i be the variable cost on production
for 1 unit of car i

Objective: $\max \sum_i^3 r_i u_i p_i - \sum_i^3 v_i u_i p_i$

$\sum_i^3 st_i u_i \leq 600$

$\sum_i^3 lb_i u_i \leq 60000$

$70 p_i \leq u_i \leq M p_i$

M: large number = GRB.INFINITY

Objective Function to max profit

$\Rightarrow \max (11 u_s p_s + 14 u_v p_v + 16 u_t p_t) - (8 u_s p_s + 10 u_v p_v + 11 u_t p_t)$

s.t. $1.5 u_s + 3 u_v + 5 u_t \leq 600$

$280 u_s + 250 u_v + 320 u_t \leq 60000$

$70 p_s \leq u_s$

$u_s \leq M p_s$

$70 p_v \leq u_v$

$u_v \leq M p_v$

$70 p_t \leq u_t$

$u_t \leq M p_t$

M here = large number like GRB.INFINITY to constrain u_i to be 0 in case the corresponding p_i is 0, and atleast 70 otherwise.

constraints to limit steel, labour usage and for atleast 70 cars in case car i is produced.

Problem Formulation

Optimal Solution to Problem Instance

Objective value : in K\$ 854 = \$854000 — profit in \$

No. of Sprint cars produced (u_s) = 70

No. of voyager cars produced (u_v) = 161

No. of titan cars produced (u_t) = 0

— cars of each type produced optimally.

Production of sprint cars $\{0,1\}$ (P_s) = 1

Production of voyager cars $\{0,1\}$ (P_v) = 1

Production of titan cars $\{0,1\}$ (P_t) = 0

— if cars of each type are produced (1) or not (0).

Problem 3

Let st_i be the amount of steel required for 1 unit of car i — $i \in \{$sprint (s), voyager (v), titan (t)$\}$

Let lb_i be the amount of labor required for 1 unit of car i

Let u_i be the number of units of car i produced. $\rightarrow u_s, u_v, u_t \rightarrow$ decision variables

Let c_i be the binary variable $\{0,1\}$ to indicate if case i for production occurs $\rightarrow c_s, c_v, c_t, c_{sv}, c_{st}, c_{vt}, c_{svt}$

cases $\hookrightarrow$ these indicate sprint only, voyager only, titan only, sprint and voyager, sprint and titan, voyager and titan, sprint, titan and voyager (all 3) respectively

Let r_i be the revenue on selling for 1 unit of car i

Let v_i be the variable cost on production for 1 unit of car i

Let f_i be the fixed cost in each production case i, $i \in \{s, v, t, sv, st, vt, svt\}$.

Objective: $\max \sum_i c_i \left(\sum_i r_i u_i - \sum_i v_i u_i - f_i \right)$

$\hookrightarrow$ i depends on c_i and varies for each case as seen on the right

s.t. $\sum_i^3 st_i u_i \leq 600$

$\sum_i^3 lb_i u_i \leq 60000$

$\sum_i^7 c_i = 1$.

Objective Function to max profit

$\Rightarrow \max\ c_s (11 u_s - 8 u_s - 180) + c_v (14 u_v - 10 u_v - 240) + c_t (16 u_t - 11 u_t - 240) +$
$c_{sv} (11 u_s + 14 u_v - 8 u_s - 10 u_v - 300) + c_{vt} (14 u_v + 16 u_t - 10 u_v - 11 u_t - 400) +$
$c_{st} (11 u_s + 16 u_t - 8 u_s - 11 u_t - 420) + c_{svt} (11 u_s + 14 u_v + 16 u_t - 8 u_s - 10 u_v - 11 u_t - 450)$

s.t. $1.5 u_s + 3 u_v + 5 u_t \leq 600$

$280 u_s + 250 u_v + 320 u_t \leq 60000$

$c_s + c_v + c_t + c_{sv} + c_{st} + c_{vt} + c_{svt} = 1$ $\rightarrow$ to pick only one case

constraints to limit steel, labour usage in each case, and also to pick only one case.

Problem Formulation

Optimal Solution to Problem Instance

Objective value : in K\$ 564 = \$564000 — profit in \$

No. of Sprint cars produced (u_s) = 64

No. of voyager cars produced (u_v) = 168

No. of titan cars produced (u_t) = 0

— cars of each type produced optimally.

Production case c_s = 0

Production case c_v = 0

Production case c_t = 0

Production case c_{sv} = 1

— if case i is the one that occurs in production (1) or not (0).

Production case Cv = 0
Production case Ct = 0
Production case Csv = 1
Production case Cvt = 0
Production case Cst = 0
Production case Csvt = 0

if case i is the one that occurs in production (1) or not (0).

Problem 4

7. A job shop has 4 jobs (A, B, C & D) that must be processed on a single machine. The processing times are respectively 2, 4, 6 and 8 days. The due dates for each job are respectively 4, 14, 10 and 16 days from now. For each day a job is late the shop must compensate the customer $100. Use dynamic programming to determine how the jobs should be sequenced to minimize the total overdue penalty costs.

Job	Processing	Due
A	2	4
B	4	14
C	6	10
D	8	16

information provided.

Stages: Slots $n = 1,2,3,4$
Decisions: x_n is the job for slot $n \in \{A,B,C,D\}$
States: S_n ~ jobs that remain to be assigned.

Dynamic Programming Inputs
Used in the model written below

$S_4 \in \{A,B,C,D\}$
$S_3 \in \{AB, AC, AD, BC, BD, CD\}$
$S_2 \in \{ABC, ABD, ACD, BCD\}$
$S_1 \in \{ABCD\}$

All possible states

Dynamic Model.

explaining states:

Stage $n=4$	Decision $x_4^*(S_4)$
$F_4(A) = 1600$	A
$F_4(B) = 600$	B
$F_4(C) = 1000$	C
$F_4(D) = 400$	D ✓ pick D as min penalty

$F_4(x)$ = no. of late days x \$100 if job x is done based on the due date in the table.

eg. If A is completed on day 20 but is due on day 4, we are 16 days late = \$100 x 16 = 1600.

Stage $n=3$ — eg. days taken already for C, D = 6+8 = 14

Stage $n=3$	Decision $x_3^*(S_3)$
$F_3(AB) = \min_{AB}[1200+600,\ 400+1600] = 1800$ (14)	A
$F_3(AC) = \min_{AC}[1000+1000,\ 800+1600] = 2000$ (12)	A
$F_3(AD) = \min_{AD}[800+400,\ 200+1600] = 1200$ (10)	A
$F_3(BC) = \min_{BC}[0+1000,\ 600+600] = 1000$ (10)	B
$F_3(BD) = \min_{BD}[0+400,\ 0+600] = 400$ (8)	B ✓ pick B as min penalty
$F_3(CD) = \min_{CD}[200+400,\ 0+1000] = 600$ (6)	C

$f_3(xy)$ = min no. of late days x \$100 if job x or y is done based on the due date in the table.

eg. If A is completed assuming B is the other possibility, it will be completed on day 16 (6 for C, 8 for D, 2 for A), and thus would be 12 days late = 1200 + 600 from B being done in stage 4. We pick the min of this way or if B was completed in stage 3, A in stage 4.

Stage $n=2$	Decision $x_2^*(S_2)$
$F_2(ABC) = \min_{ABC}[600+1000,\ 0+2000,\ 400+1800] = 1600$ (8)	A
$F_2(ABD) = \min_{ABD}[400+400,\ 0+1200,\ 0+1800] = 800$ (6)	A
$F_2(ACD) = \min_{ACD}[200+600,\ 0+1200,\ 0+2000] = 800$ (4)	A
$F_2(BCD) = \min_{BCD}[0+600,\ 0+400,\ 0+1000] = 400$ (2)	C ✓ pick C as min penalty

$f_3(xyz)$ = min no. of late days x \$100 if job x, y or z is done based on the due date in the table.

eg. similar method as above with all possible scenarios.

$$f_2(BCD) = \min_{BCD}^{ACD} [0+600, 0+400, 0+1000] = 400$$ C ✓ pick C as min penalty

Stage | Decision
n=1 | $X_i^*(S_i)$

$$f_1(ABCD) = \min_{ABCD} [0+400, 0+800, 0+800, 0+1600] = 400$$ A ✓ pick A as min penalty

$f_4(xyzw)$ = min no. of late days x $100 if job x, y, z or w is done based on the due date in the table.

eg. checking each job and picking the minimum penalty cost.

Problem Instance Solution

min penalty = 400$] min overdue penalty as seen above.

A→C→B→D] optimal ordering sequence

Stages	1 →	2 →	3 →	4
Job	A →	C →	B →	D
Due	4	10	14	16
Processing	2	2+6	2+6+4	2+6+4+8
Penalty	0	0	0	4 ↳ $100 x 4 = $400

] stagewise Job decision.

] Solution visualized.