



ΕΠΙΧΕΙΡΗΣΙΑΚΗ ΕΡΕΥΝΑ Project OR-2023

Δεϊρμεντζόγλου Ιωάννης ΑΕΜ—10015 deirmentz@ece.auth.gr

Τα 2 προβλήματα μοντελοποιήθηκαν σε AMPL και ως solver χρησιμοποιήθηκε ο Gurobi

Πρόβλημα 1

Μεταβλητές :

- **rank (Sequence, Component)**: binary variable οπού είναι 1 όταν το εξάρτημα Component είναι στην θέση Sequence, 0 αλλιώς.
- **completionTime (Sequence)**: συνολικός χρόνος ολοκλήρωσης της επεξεργασίας του εξαρτήματος στην θέση Sequence.
- **Time (Sequence, Procedures)**: ο χρόνος επεξεργασίας για το εξάρτημα στην θέση Sequence και διαδικασία Procedure
- **Wait (Sequence, Procedures)**: ο χρόνος αναμονής για το εξάρτημα Sequence πριν μπει στην διαδικασία Procedure.

Μοντελοποίηση

Objective function: Συνολικός χρόνος του εξαρτήματος στην θέση 6

minimize: CompletionTime [6]

Constraints:

1. $\sum_{iC=1}^6 rank(iS, iC) = 1$

Μόνο ένα εξάρτημα στην θέση iS.

2. $\sum_{iS=1}^6 rank(iS, iC) = 1$

Μόνο σε μια θέση το κάθε εξάρτημα.

3. $\sum_{iP=1}^3 wait(1, iP)=0$

Το εξάρτημα που μπαίνει 1^ο δεν περιμένει σε κάποια διαδικασία γιατί το σύστημα είναι άδαιο επομένως ο χρόνος αναμονής είναι 0 .

4. $\sum_{iS=2}^6 wait(iS, 'Casting')$

Πρόκειται για τον χρόνο αναμονής του εξαρτήματος πριν μπει στην διαδικασία της χύτευσης. Ο χρόνος αναμονής θα είναι ο χρόνος επεξεργασίας (χύτευσης) όλων των εξαρτημάτων που προηγούνται του εξαρτήματος στην θέση iS .

5. $Time = \sum_{iC=1}^6 rank(iC, iS) * ProcessTime(Procedure, iC)$

Είναι ο χρόνος επεξεργασίας που το εξάρτημα στην θέση iS και διαδικασία Procedure .

6. $\sum_{iS=2}^6 wait(iS, 'Grinding')$

Πρόκειται για τον χρόνο αναμονής του εξαρτήματος πριν μπει στην διαδικασία της λείανσης. Αυτός ο χρόνος είναι μη μηδενικός μόνο όταν το άθροισμα των wait σε casting και grinding και των χρόνων επεξεργασίας του προηγούμενου εξαρτήματος είναι μεγαλύτερος του $wait(iS, 'casting') + TimeCasting(iS)$. Δηλαδή, πιο απλά μόνο όταν ενώ έχει βγει το εξάρτημα από την χύτευση το προηγούμενο δεν έχει βγει από την λείανση .

7. $\sum_{iS=2}^6 wait(iS, 'Connection')$

Παρόμοια λογική με το προηγούμενο απλά για την διαδικασία της σύνδεσης και τον χρόνο αναμονής για να μπει το εξάρτημα στην σύνδεση

8. completionTime

Αθροίζοντας τον χρόνο αναμονής πριν από κάθε διαδικασία και τον χρόνο επεξεργασίας υπολογίζω τον συνολικό χρόνο ολοκλήρωσης για κάθε εξάρτημα. Για το τελευταίο εξάρτημα ο χρόνος ολοκλήρωσης είναι και ο συνολικός χρόνος κατασκευής .

ΑΠΟΤΕΛΕΣΜΑΤΑ

Ο ελάχιστος συνολικός χρόνος επεξεργασίας είναι: 35 ώρες

$$\text{TotalTime} = 35$$

Η σειρά με την οποία θα ξεκινήσουν την επεξεργασία τα εξαρτήματα είναι (για ευκολία θα αντικαταστήσω το 1^ο με A ,το 2^ο με B κλπ.) :

C , A , D , F , E , B

rank	[*,*]					
:	1	2	3	4	5	6
A	0	1	0	0	0	0
B	0	0	0	0	0	1
C	1	0	0	0	0	0
D	0	0	1	0	0	0
E	0	0	0	0	1	0
F	0	0	0	1	0	0

Display Rank

Πρόβλημα 2

Sets:

```
set ProdLines := Basic ShowOff Luxurious NewBasic NewShowOff ;
set Models    := BasicM ShowOffM LuxuriousM ;
set ModelsLines:= B_B SO_SO L_L B_NB SO_NB B_NS0 SO_NS0 L_NS0 ;
```

Μεταβλητές :

- **x(ProductionLines)**: binary μεταβλητή που είναι 1 όταν η γραμμή είναι η ProductionLines
- **numCars (ModelsLines)**: αριθμός αυτοκινήτων που παράγεται από το κάθε μοντέλο σε κάθε γραμμή παραγωγής
- **diffBasic**: διαφορά μεταξύ ζήτησης και παραγωγής για το μοντέλο Basic
- **diffShowOff**: διαφορά μεταξύ ζήτησης και παραγωγής για το μοντέλο Show Off

- **diffLuxurious**: διαφορά μεταξύ ζήτησης και παραγωγής για το μοντέλο Luxurious

Μοντελοποίηση

Objective function: Θέλω να μεγιστοποιήσω το κέρδος της βιομηχανίας. Το συνολικό κέρδος θα είναι το κέρδος ανά μοντέλο και γραμμή παραγωγής επί τον αριθμό αυτοκινήτων που παράγεται από το κάθε μοντέλο σε κάθε γραμμή παραγωγής μείον το σταθερό κόστος κάθε γραμμής .

maximize:

```
maximize TotalProfit :
  Profit['BasicM','Basic']*numCars['B_B'] + Profit['ShowOffM','ShowOff']*numCars['SO_SO']
+ Profit['LuxuriousM','Luxurious']*numCars['L_L'] + Profit['BasicM','NewBasic']*numCars['B_NB']
+ Profit['ShowOffM','NewBasic']*numCars['SO_NB'] + Profit['BasicM','NewShowOff']*numCars['B_NSO']
+ Profit['ShowOffM','NewShowOff']*numCars['SO_NSO'] + Profit['LuxuriousM','NewShowOff']*numCars['L_NSO']
- sum{ic in ProdLines}{fixedCost[ic]*x[ic]} ;
```

Constraints:

1. **Basic or New Basic: $x['Basic'] + x['New Basic'] = 1$**
Η γραμμή μπορεί να εκμοντερνιστεί ή όχι .
2. **Show Off or New Show Off: $x['ShowOff'] + x['NewShowOff'] = 1$**
Η γραμμή μπορεί να εκμοντερνιστεί ή όχι .
3. **$x['Luxurious'] = 1$**
Η γραμμή Luxurious δεν μπορεί να εκμοντερνιστεί.
4. **$numCars['B_B'] \leq x['Basic'] * ProdCapality['Basic']$**
Ο αριθμός Basic μοντέλων από την γραμμή Basic θα είναι μικρότερος ή ίσος της δυνατότητα παραγωγής της γραμμής μόνο εάν η γραμμή δεν έχει εκμοντερνιστεί.
5. **$numCars['SO_SO'] \leq x['ShowOff'] * (ProdCapability['ShowOff'] - DemandChange['BasicM','ShowOffM']*diffBasic)$**
Ο αριθμός των Showoff μοντέλων από την γραμμή Show Off θα είναι θα είναι μικρότερος ή ίσος της δυνατότητας παραγωγής μείον την μεταβολή της ζήτησης από τα Basic που δεν καλύπτουν την ζήτηση για Basic (diffBasic) . Όλα αυτά εφόσον δεν έχει εκμοντερνιστεί η γραμμή Show Off .

```

numberCars_L_L :
    numCars['L_L'] <= x['Luxurious'] * ( ProdCapability['Luxurious'] - DemandChange['BasicM','LuxuriousM']*diffBasic
    - DemandChange['ShowOffM','LuxuriousM']*diffShowOff ) ;

```

6. Ο αριθμός των Luxurious μοντέλων από την γραμμή Luxurious θα είναι μικρότερος ή ίσος της δυνατότητας παραγωγής μείον την μεταβολή της ζήτησης από τα Basic και Show Off που δεν καλύπτουν τα 2 μοντέλα Basic και ShowOff. Όλα αυτά εφόσον δεν έχει εκμοντερνιστεί η γραμμή Luxurious .

```

numberCars_B_SO_NB :
    numCars['B_NB'] + numCars['SO_NB'] <= x['NewBasic']*ProdCapability['NewBasic']

```

7. Ο αριθμός των Basic και Show Off μοντέλων από την γραμμή New Basic είναι μικρότερος ή ίσος της δυνατότητας παραγωγής αν έχει εκμοντερνιστεί η γραμμή New Basic .

```

numberCars_B_SO_L_NSO :
    numCars['B_NSO'] + numCars['SO_NSO'] + numCars['L_NSO']
    <= x['NewShowOff'] * ( ProdCapability['NewShowOff']- DemandChange['ShowOffM','LuxuriousM']*diffShowOff
    - DemandChange['BasicM','LuxuriousM']*diffBasic - DemandChange['BasicM','ShowOffM']*diffBasic ) ;

```

8. Ο αριθμός των μοντέλων Basic, Show Off, Luxurious από την γραμμή NewShowOff είναι μικρότερος η ίσος της δυνατότητας παραγωγής μείον τις μεταβολές της ζήτησης για κάθε μοντέλο, τα οποία δεν καλύπτονται από την ζήτηση.

```

DemandBasic :
    x['Basic']*numCars['B_B'] + x['NewBasic']*numCars['B_NB'] + x['NewShowOff']*numCars['B_NSO']
    + diffBasic = 1400 ;

```

9. Η ζήτηση για τα μοντέλα Basic

```

DemandShowOff :
    x['ShowOff']*numCars['SO_SO'] + x['NewBasic']*numCars['SO_NB'] + x['NewShowOff']*numCars['SO_NSO']
    + DemandChange['BasicM','ShowOffM']*diffBasic + diffShowOff = 1100 ;

```

10. Η ζήτηση για τα μοντέλα Show Off

```

DemandLuxurious :
    x['Luxurious']*numCars['L_L'] + x['NewShowOff']*numCars['L_NSO'] + DemandChange['BasicM','LuxuriousM']*diffBasic
    + DemandChange['ShowOffM','LuxuriousM']*diffShowOff = 800 ;

```

11. Η ζήτηση για τα μοντέλα Luxurious

ΑΠΟΤΕΛΕΣΜΑΤΑ

Για την επίλυση χρησιμοποιήθηκε ο solver της gurobi

Για την μεγιστοποίηση του κέρδους συμβουλευόμαστε την εταιρεία να έχει τις εξής γραμμές : **Basic** , **NewShowOff** , **Luxurious**

```
Basic 1
Luxurious 1
NewBasic 0
NewShowOff 1
ShowOff 0
```

Display binary x{ProdLines}

Για να καλυφθεί η ζήτηση θα πρέπει να παραχθούν :

- Από την γραμμή **Basic** : **700** (χιλιάδες μονάδες αυτοκινήτων) από τα **Basic** μοντέλα .
- Από την γραμμή **NewShowOff** : **700.000** από τα **Basic** μοντέλα , **1.100.000** από τα **Show Off** .
- Από την γραμμή **Luxurious** : **800.000** από τα **Luxurious** μοντέλα .

```
B_B 700
B_NB 0
B_NSO 700
L_L 800
L_NSO 0
SO_NB 0
SO_NSO 1100
SO_SO 0
```

Display numCars{ModelsLines}

Καλύπτεται όλη η ζήτηση

```
diffBasic = 0
diffShowOff = 0
diffLuxurious = 0
```

Το συνολικό κέρδος θα είναι: **2.560.000 ευρώ**

$$\text{TotalProfit} = 2560$$

ΠΑΡΑΤΗΡΗΣΗ

Εάν η εταιρεία δεν είναι υποχρεωμένη να έχει την γραμμή παραγωγής Luxurious παίρνουμε την εξής λύση :

Οι γραμμές που θα έχω είναι : **NewBasic , New ShowOff**

```

Basic      0
Luxurious  0
NewBasic   1
NewShowOff 1
ShowOff    0

```

Display binary x{ProdLines}

Για να καλυφθεί η ζήτηση θα πρέπει να παραχθούν :

- Από την γραμμή **NewBasic** : **1.400.000** από τα **Basic** μοντέλα και **1.000.000** από **ShowOff**.
- Από την γραμμή **NewShowOff** : **1.000.000** από τα **Show Off** και **8.000.000** από τα **Luxurious** .

```

B_B      0
B_NB    1400
B_NSO    0
L_L      0
L_NSO    800
SO_NB    100
SO_NSO  1000
SO_SO    0

```

Display numCars {ModelsLines}

Καλύπτεται όλη η ζήτηση

```

diffBasic = 0
diffShowOff = 0
diffLuxurious = 0

```

Το συνολικό κέρδος θα είναι: **4.040.000 ευρώ**

$$\text{TotalProfit} = 4040$$

