

Επιχειρησιακή Έρευνα

Ενότητα 2

Βελτιστοποίηση με ΗΥ - AMPL

Αθανασία Μάνου

Διαπανεπιστημιακό Διατμηματικό
Μεταπτυχιακό Πρόγραμμα Σπουδών
Μαθηματικά της Αγοράς και της Παραγωγής

Η γλώσσα AMPL

- Η γλώσσα AMPL (A Mathematical Programming Language) έχει σχεδιαστεί ειδικά για την επίλυση προβλημάτων βελτιστοποίησης.
- Επιτρέπει την εισαγωγή στον Η/Υ ενός προβλήματος βελτιστοποίησης σε μορφή πολύ κοντινή στη συνήθη μαθηματική μορφή.
- Συνεργάζεται με πολλούς αλγόριθμους επίλυσης.
- Συνεργάζεται με πολλούς τρόπους εισαγωγής των δεδομένων.
- Παρέχει πολλούς τρόπους διαχείρισης των αποτελεσμάτων.

Πηγές για την AMPL

- Ο ιστότοπος της AMPL βρίσκεται στο `http://www.ampl.com`
- Για να κατεβάσουμε δοκιμαστική έκδοση της AMPL επιλέγουμε
 - start for free
 - academics ή enthusiasts
 - Κάνουμε εγγραφή και είσοδο
 - Download
- Για το πλήρες εγχειρίδιο της AMPL πηγαίνουμε στο `resources -> optimization books -> The AMPL book`

“Μικρό” παράδειγμα προγραμματισμού παραγωγής

- Ελαιοτριβείο παράγει δυο τύπους ελαιολάδου, τον “κλασικό” (εξευγενισμένο) και τον “παρθένο”.
- Το ελαιοτριβείο έχει διαθέσιμες 40 ώρες παραγωγής ανά εβδομάδα.
- 1 λίτρο κλασικού πωλείται 10 ευρώ.
1 λίτρο παρθένου πωλείται 15 ευρώ.
- Το ελαιοτριβείο μπορεί να παράγει σε κάθε δεδομένη στιγμή μόνο έναν τύπο ελαιολάδου.
- Σε 1 ώρα μπορεί να παράγει 40 λίτρα κλασικού ή 30 λίτρα παρθένου.
- Το εμπορικό τμήμα του ελαιοτριβείου εκτιμά ότι μπορεί να πουλήσει το πολύ 1000 λίτρα κλασικού και 860 λίτρα παρθένου κάθε εβδομάδα.
- Να μεγιστοποιηθεί ο συνολικός τζίρος ανά εβδομάδα.

Μεταβλητές απόφασης

O_{ilc} : ποσότητα (σε l) από το χυμό πορτοκάλι

O_{ilv} : >> >> >> νερό >>

Πρόβλημα

$$\max 10 \cdot O_{ilc} + 15 \cdot O_{ilv}$$

$$\text{υπό } \frac{1}{40} O_{ilc} + \frac{1}{30} O_{ilv} \leq 40$$

$$0 \leq O_{ilc} \leq 1000$$

$$0 \leq O_{ilv} \leq 860$$

Μοντελοποίηση ως π.γ.π.

- Oil_c, Oil_v : Οι ποσότητες κλασικού και παρθένου ελαιολάδου που θα παραχθούν σε μια εβδομάδα.
- Έχουμε το π.γ.π.

$$\begin{aligned} \max \quad & 10Oil_c + 15Oil_v \\ \text{υπό} \quad & \frac{1}{40}Oil_c + \frac{1}{30}Oil_v \leq 40 \\ & 0 \leq Oil_c \leq 1000 \\ & 0 \leq Oil_v \leq 860. \end{aligned}$$

Μοντελοποίηση στην AMPL

- Το π.γ.π.

$$\begin{aligned} \max \quad & 10Oil_c + 15Oil_v \\ \text{υπό} \quad & \frac{1}{40}Oil_c + \frac{1}{30}Oil_v \leq 40 \\ & 0 \leq Oil_c \leq 1000 \\ & 0 \leq Oil_v \leq 860. \end{aligned}$$

γράφεται σε έναν text editor ως

```
## Example 01
var Oil_c; # amount of classic
var Oil_v; # amount of virgin
maximize profit: 10*Oil_c + 15*Oil_v;
subject to time: (1/40)*Oil_c + (1/30)*Oil_v <= 40;
subject to classic_limit: 0 <= Oil_c <= 1000;
subject to virgin_limit: 0 <= Oil_v <= 860;
```

Βασική σύνταξη στην AMPL

```
## Example 01  
var Oil_c; # amount of classic  
var Oil_v; # amount of virgin  
maximize profit: 10*Oil_c + 15*Oil_v;  
subject to time: (1/40)*Oil_c + (1/30)*Oil_v <= 40;  
subject to classic_limit: 0 <= Oil_c <= 1000;  
subject to virgin_limit: 0 <= Oil_v <= 860;
```

- # → Αρχή σχολίου.
- var → Δήλωση μεταβλητής.
- ; → Τέλος γραμμής εντολής.
- maximize ή minimize + όνομα + : → Αντικ. συναρτ.
- subject to + όνομα + : → Περιορισμός.
- Case sensitive ονόματα. Τα ονόματα είναι μοναδικά.

Εκτέλεση, διαχείριση αρχείων στην AMPL

- Ένα αρχείο που περιέχει ένα πρόβλημα-μοντέλο πρέπει να σώζεται με την επέκταση `.mod`.
Π.χ. το προηγούμενο θα μπορούσε να σωθεί σε κάποια θέση ως `Example01.mod`.
- Για να εκτελεστεί ένα αρχείο πρέπει να τρέξουμε την εφαρμογή AMPL και να εμφανιστεί η αναμονή `ampl:.`
- Επιλέγουμε αλγόριθμο επίλυσης (solver).
Π.χ. `option solver cplex;.`
- Φορτώνουμε το πρόβλημα-μοντέλο.
Π.χ. `model Example01.mod;.`
- Επιλύουμε με την εντολή `solve;.`

Διορθώσεις και εμφάνιση λύσης

- Αν υπάρξει λάθος στο μοντέλο κατά το τρέξιμο του αλγορίθμου επίλυσης τότε
 - Διορθώνουμε το `.mod` αρχείο στον text editor.
 - Δίνουμε την εντολή `reset;` στο AMPL.
 - Ξαναφορτώνουμε το μοντέλο.
- Η εντολή `solve;` θα μας δώσει κάποιες πληροφορίες και τη βέλτιστη τιμή της αντικειμενικής. Π.χ.
CPLEX 6.5.3: optimal solution; objective 17433.33333
2 simplex iterations (0 in phase I)
- Οι τιμές των μεταβλητών στη βέλτιστη λύση εμφανίζονται με την εντολή `display + όνομα + ;`.
Π.χ. `display Oil_c;`
και `display Oil_v;`

Αποθήκευση εξόδου σε αρχείο

- Για να αποθηκεύσουμε την έξοδο σε ένα αρχείο έχουμε δυο δυνατότητες, να δημιουργήσουμε-αντικαταστήσουμε (over-write) ένα αρχείο ή να προσθέσουμε περιεχόμενο σε ένα ήδη υπάρχον αρχείο.
- Π.χ. με την εντολή

```
display Oil_c > Example01.out;
```

θα δημιουργηθεί αρχείο με το όνομα Example01.out .
Αν υπάρχει αρχείο με τέτοιο όνομα θα αντικατασταθεί.

- Με την εντολή

```
display Oil_c >> Example01.out;
```

θα προστεθεί στο αρχείο με το όνομα Example01.out η μεταβλητή Oil_c με την τιμή της.

Γενίκευση παραδ. προγραμματισμού παραγωγής

- Ελαιοτριβείο παράγει n τύπους ελαιολάδου.
- Το ελαιοτριβείο έχει διαθέσιμες t ώρες παραγωγής ανά εβδομάδα.
- 1 λίτρο τύπου i πωλείται p_i ευρώ.
- Το ελαιοτριβείο μπορεί να παράγει σε κάθε δεδομένη στιγμή μόνο έναν τύπο ελαιολάδου.
- Σε 1 ώρα μπορεί να παράγει r_i λίτρα τύπου i .
- Το εμπορικό τμήμα του ελαιοτριβείου εκτιμά ότι μπορεί να πουλήσει το πολύ m_i λίτρα τύπου i κάθε εβδομάδα.
- Να μεγιστοποιηθεί ο συνολικός τζίρος ανά εβδομάδα.

Μεταβλητές απόφασης:

x_i = ποσότητα (σε λίτρα) να θα πουλάμε από τον χώρο i ,
 $i=1,2,\dots,n$.

Προβλήματα

$$\max \sum_{i=1}^n p_i x_i$$

$$\text{υπό} \sum_{i=1}^n (1/r_i) x_i \leq t$$

$$0 \leq x_i \leq m_i, \quad i=1,2,\dots,n$$

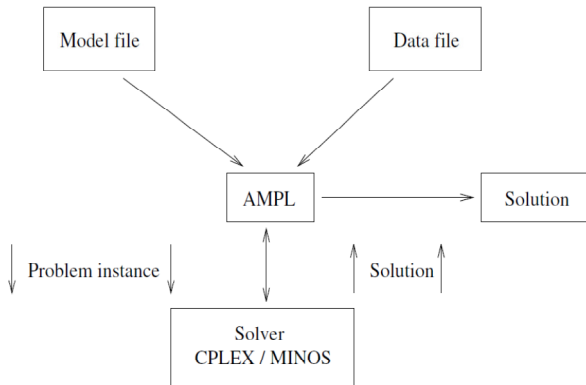
Μοντελοποίηση ως π.γ.π.

- x_i : Η ποσότητα ελαιολάδου τύπου i που θα παράγεται σε μια εβδομάδα.
- Έχουμε το π.γ.π.

$$\begin{array}{ll}\max & \sum_{i=1}^n p_i x_i \\ \text{υπό} & \sum_{i=1}^n (1/r_i) x_i \leq t \\ & 0 \leq x_i \leq m_i, \quad i = 1, 2, \dots, n.\end{array}$$

- Για $n = 2$, $t = 40$, $p_1 = 10$, $p_2 = 15$, $m_1 = 1000$ και $m_2 = 860$ παίρνουμε το μικρό πρόβλημα παραγωγής.
- Το μικρό πρόβλημα παραγωγής είναι μια ειδική περίπτωση μοντέλου και όχι ένα μοντέλο.
- Το AMPL ενθαρρύνει το διαχωρισμό μοντέλου από τα δεδομένα και τον αλγόριθμο επίλυσης.

AMPL: Μοντέλο, δεδομένα, λύση



- Διάγραμμα αλληλεπίδρασης στοιχείων του AMPL

Μοντελοποίηση στην AMPL

- Το π.γ.π.

$$\begin{array}{ll}\max & \sum_{i=1}^n p_i x_i \\ \text{υπό} & \sum_{i=1}^n (1/r_i) x_i \leq t \\ & 0 \leq x_i \leq m_i, \quad i = 1, 2, \dots, n.\end{array}$$

γράφεται ως

```
## Example 02 - version 1
```

```
param n;
```

```
param t;
```

```
param p{i in 1..n};
```

```
param r{i in 1..n};
```

```
param m{i in 1..n};
```

```
var x{i in 1..n};
```

```
maximize profit: sum{i in 1..n} p[i]*x[i];
```

```
subject to time: sum{i in 1..n} (1/r[i])*x[i] <= t;
```

```
subject to capacity{i in 1..n}: 0 <= x[i] <= m[i];
```

Χρησιμοποιώντας δείκτες στην AMPL

```
## Example 02 - version 1
param n;
param t;
param p{i in 1..n};
param r{i in 1..n};
param m{i in 1..n};
var x{i in 1..n};
maximize profit: sum{i in 1..n} p[i]*x[i];
subject to time: sum{i in 1..n} (1/r[i])*x[i] <= t;
subject to capacity{i in 1..n}: 0 <= x[i] <= m[i];
```

- `param` → Δήλωση παραμέτρου.
- `{i in 1..n}` → Δήλωση δείκτη και συνόλου διαδοχικών ακεραίων στους οποίους κινείται.
- Ένας δείκτης μπορεί να αφορά παραμέτρους, μεταβλητές, αθροίσεις ή περιορισμούς.

Δεδομένα στην AMPL I

```
## Data 1 for Example 02 - version 1  
param n:= 2;  
param t:= 40;  
param p:= 1 10 2 15;  
param r:= 1 40 2 30;  
param m:= 1 1000 2 860;
```

- `param + όνομα + :=` → Δήλωση τιμών παραμέτρου.
- Αν μια παράμετρος είναι διανυσματική, τότε γράφουμε το δείκτη κάθε συνιστώσας και μετά την τιμή της.
- Το AMPL αγνοεί κενά και αλλαγές γραμμών (carriage returns).

Δεδομένα στην AMPL II

```
## Data 1 for Example 02 - version 1
```

```
param n:= 2;
```

```
param t:= 40;
```

```
param p:= 1 10 2 15;
```

```
param r:= 1 40 2 30;
```

```
param m:= 1 1000 2 860;
```



```
## Data 2 for Example 02 - version 1
```

```
param n:= 2;
```

```
param t:= 40;
```

```
param p:= 1 10
```

```
2 15;
```

```
param r:= 1 40
```

```
2 30;
```

```
param m:= 1 1000
```

```
2 860;
```

Δεδομένα στην AMPL III

```
## Data 2 for Example 02 - version 1
```

```
param n:= 2;
```

```
param t:= 40;
```

```
param p:= 1 10  
          2 15;
```

```
param r:= 1 40  
          2 30;
```

```
param m:= 1 1000  
          2 860;
```



```
## Data 3 for Example 02 - version 1
```

```
param n:= 2;
```

```
param t:= 40;
```

```
param:      p    r    m:=  
          1 10 40 1000  
          2 15 30 860;
```

AMPL Μοντέλο, δεδομένα, επίλυση

- Τυπική ακολουθία εντολών για την εισαγωγή μοντέλου, δεδομένων, την επίλυση και την εμφάνιση της λύσης:

```
reset;  
model Example02v1.mod;  
data Example02v1d1.dat;  
solve;  
display x;
```