Ειδικά Θέματα Επιχειρησιακής Έρευνας
Ενότητα 3
Εισαγωγή στον Ακέραιο Προγραμματισμό

Γιάννης Δημητρακόπουλος

Τμήμα Μαθηματικών
Πανεπιστήμιο Αθηνών
Δ.Π.Μ.Σ. Μαθηματικά της αγοράς και της Παραγωγής

8 Οκτωβρίου 2020

Προβλήματα ακέραιου προγραμματισμού

- Πρόβλημα (γραμ.) ακέραιου προγραμματισμού (π.α.π.) = π.γ.π. + Περιορισμός κάποιων μεταβλητών στους ακεραίους.
- Τυπική μορφή π.α.π. :

$$
\begin{array}{llllllllll}
\max & c_1x_1 & + & c_2x_2 & + & \cdots & + & c_nx_n & & \\
\text{υπό} & a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & \leq & b_1 \\
& a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & \leq & b_2 \\
& & & & & \vdots & & & & \\
& a_{m1}x_1 & + & a_{m2}x_2 & + & \cdots & + & a_{mn}x_n & \leq & b_m \\
& \multicolumn{9}{l}{x_1, x_2, \ldots, x_n \geq 0.} \\
& \multicolumn{9}{l}{x_i \text{ ακέραιος}, \ i \in I.}
\end{array}
$$

Χαλάρωση γραμμικού προγραμματισμού

- Χαλάρωση γραμμικού προγραμματισμού (χ.γ.π.) ενός π.α.π.
= π.γ.π. με ίδια αντικειμενική και ίδιους περιορισμούς, αλλά χωρίς τους περιορισμούς ακεραιότητας.
- Η εφικτή περιοχή της χ.γ.π. ενός π.α.π. είναι υπερσύνολο της εφικτής περιοχής του π.α.π.
- Αν η βέλτιστη λύση του χ.γ.π. ενός π.α.π. ικανοποιεί τους περιορισμούς ακεραιότητας του π.α.π. τότε είναι βέλτιστη λύση του π.α.π.

Δυαδικές αποφάσεων

- Μία δυαδική απόφαση αφορά
  να κάνω το A ή να μην το κάνω και μοντελοποιείται ως εξής:
  Μεταβλητή Απόφασης:

$$x_A = \begin{cases} 1, & \text{αν ληφθεί η απόφαση } A \\ 0, & \text{διαφορετικά.} \end{cases}$$

Μοντελοποίηση με Δυαδικές Μεταβλητές

- Μοντελοποίηση δυαδικών αποφάσεων.
- Μοντελοποίηση εξαρτώμενων αποφάσεων (contigent decisions).
- Μοντελοποίηση Αμοιβαίως Αποκλειόμενων Αποφάσεων (mutually exclusive decisions)
- Μοντελοποίηση διαζευκτικών περιορισμών.
- Μοντελοποίηση μεταβλητών με πεπερ. πλήθος τιμών.
- Μοντελοποίηση κατά τμήματα γραμμικών συναρτήσεων.

Παράδειγμα - Δυαδικές αποφάσεις

- Επέκταση κατασκευαστικής εταιρείας.
- Κτίσιμο νέων εργοστασίων, αποθηκών σε Αθήνα/Θεσ.
- Αποφάσεις - Απαιτούμενα κεφάλαια - Αποδόσεις:

α/α	Απόφαση	Απαιτ. κεφάλαιο	Απόδοση
1	Εργοστ. Αθήνα	6 εκατ.	9 εκατ.
2	Εργοστ. Θεσ.	3 εκατ.	5 εκατ.
3	Αποθ. Αθήνα	5 εκατ.	6 εκατ.
4	Αποθ. Θεσ.	2 εκατ.	4 εκατ.

- Διαθέσιμο κεφάλαιο: 10 εκατ.
- Σκοπός: Μεγιστοποίηση απόδοσης.

Παράδειγμα - Δυαδικές αποφάσεις

- Δεδομένα: Διαθέσιμο κεφάλαιο 10 εκατ.
  Αποφάσεις - Απαιτούμενα κεφάλαια - Αποδόσεις:

α/α	Απόφαση	Απαιτ. κεφάλαιο	Απόδοση
1	Εργοστ. Αθήνα	6 εκατ.	9 εκατ.
2	Εργοστ. Θεσ.	3 εκατ.	5 εκατ.
3	Αποθ. Αθήνα	5 εκατ.	6 εκατ.
4	Αποθ. Θεσ.	2 εκατ.	4 εκατ.

- Μεταβλητές αποφάσεων:

$$x_j = \begin{cases} 1, & \text{αν ληφθεί η απόφαση } j \\ 0, & \text{διαφορετικά} \end{cases} \qquad j = 1, 2, 3, 4$$

Παράδειγμα - Δυαδικές αποφάσεις

- Δεδομένα: Διαθέσιμο κεφάλαιο 10 εκατ.
  Αποφάσεις - Απαιτούμενα κεφάλαια - Αποδόσεις:

α/α	Απόφαση	Απαιτ. κεφάλαιο	Απόδοση
1	Εργοστ. Αθήνα	6 εκατ.	9 εκατ.
2	Εργοστ. Θεσ.	3 εκατ.	5 εκατ.
3	Αποθ. Αθήνα	5 εκατ.	6 εκατ.
4	Αποθ. Θεσ.	2 εκατ.	4 εκατ.

- Π.α.π.:

$$
\begin{array}{ll}
\max & 9x_1 + 5x_2 + 6x_3 + 4x_4 \\
\text{υπό} & 6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10 \\
 & x_j \in \{0, 1\},\ j = 1, 2, 3, 4.
\end{array}
$$

Μοντελοποίηση εξαρτώμενων αποφάσεων

- Για τη μοντελοποίηση δυο δυαδικών αποφάσεων έχουν εισαχθεί οι μεταβλητές x_A και x_B.
- Αν η απόφαση A επιτρέπεται να λήφθει μόνο αν ληφθεί η απόφαση B τότε εισάγουμε τον περιορισμό

$$x_A \leq x_B.$$

Παράδειγμα - Εξαρτώμενες αποφάσεις

- Πρόβλημα επέκτασης κατασκευαστικής εταιρείας με τα ίδια δεδομένα.
- Επιπλέον περιορισμός: Η κατασκευή αποθήκης σε μια πόλη είναι δυνατή μόνο αν κατασκευαστεί σε αυτήν εργοστάσιο.
- Μεταβλητές απόφασης:
  x_1: Κατασκευή εργοστασίου στην Αθήνα.
  x_2: Κατασκευή εργοστασίου στη Θεσσαλονίκη.
  x_3: Κατασκευή αποθήκης στην Αθήνα.
  x_4: Κατασκευή αποθήκης στη Θεσσαλονίκη.
- Επιπλέον περιορισμοί:

$$
\begin{aligned}
x_3 &\leq x_1 \\
x_4 &\leq x_2.
\end{aligned}
$$

Παράδειγμα - Εξαρτώμενες αποφάσεις

- Νέο π.α.π. :

$$\begin{array}{ll} \max & 9x_1 + 5x_2 + 6x_3 + 4x_4 \\ \text{υπό} & 6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10 \\ & -x_1 + x_3 \leq 0 \\ & -x_2 + x_4 \leq 0 \\ & x_j \in \{0,1\},\ j = 1,2,3,4. \end{array}$$

Μοντελοποίηση αμοιβαίως αποκλειόμενων αποφάσεων

- Για τη μοντελοποίηση δυο δυαδικών αποφάσεων έχουν εισαχθεί οι μεταβλητές x_A και x_B.
- Αν το πολύ μία από τις αποφάσεις A, B επιτρέπεται να λήφθει τότε εισάγουμε τον περιορισμό

$$x_A + x_B \leq 1.$$

Παράδειγμα - Αμοιβαίως αποκλειόμενες αποφάσεις

- Πρόβλημα επέκτασης κατασκευαστικής εταιρείας με τα ίδια δεδομένα.
- Επιπλέον περιορισμός: Το πολύ μια αποθήκη μπορεί να κτιστεί.
- Μεταβλητές απόφασης:
  x_1: Κατασκευή εργοστασίου στην Αθήνα.
  x_2: Κατασκευή εργοστασίου στη Θεσσαλονίκη.
  x_3: Κατασκευή αποθήκης στην Αθήνα.
  x_4: Κατασκευή αποθήκης στη Θεσσαλονίκη.
- Επιπλέον περιορισμός:

$$x_3 + x_4 \leq 1.$$

- Νέο π.α.π. :

$$\begin{array}{ll} \max & 9x_1 + 5x_2 + 6x_3 + 4x_4 \\ \text{υπό} & 6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10 \\ & -x_1 + x_3 \leq 0 \\ & -x_2 + x_4 \leq 0 \\ & x_3 + x_4 \leq 1 \\ & x_j \in \{0,1\}, \;\; j = 1,2,3,4. \end{array}$$

Μοντελοποίηση διαζευκτικών περιορισμών

- Σε ένα τυπικό πρόβλημα μαθηματικού προγραμματισμού όλοι οι περιορισμοί πρέπει να ικανοποιούνται (συζεύξη - "και").
- Σε ένα μοντέλο μπορεί να υπεισέρχονται υποσύνολα περιορισμών που πρέπει εναλλακτικά να ικανοποιούνται (διάζευξη - "ή").
- Η χρήση ακέραιων μεταβλητών επιτρέπει την μοντελοποίηση διαζευκτικών (εναλλακτικών) περιορισμών.

Μοντελοποίηση διαζευκτικών περιορισμών

- Μεταβλητές αποφάσεων: $x_1, x_2, \ldots, x_n \geq 0$.
- Παράμετροι: $a_1, a_2, \ldots, a_n, c_1, c_2, \ldots, c_n, b, d \geq 0$.
- Περιορισμός: Να ισχύει τουλάχιστον μία από τις ανισότητες

$$\begin{aligned} a_1x_1 + a_2x_2 + \cdots + a_nx_n &\leq b \\ c_1x_1 + c_2x_2 + \cdots + c_nx_n &\leq d. \end{aligned}$$

- Εισάγουμε μια νέα μεταβλητή y, έναν αυθαίρετα πολύ μεγάλο θετικό αριθμό M και τους περιορισμούς

$$\begin{aligned} & a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq b + My \\ & c_1x_1 + c_2x_2 + \cdots + c_nx_n \leq d + M(1-y) \\ & y \in \{0, 1\}. \end{aligned}$$

Μοντελοποίηση διαζευκτικών περιορισμών

- Μεταβλητές αποφάσεων: $x_j \geq 0$, $j = 1, 2, \ldots, n$.
- Παράμ: $a_{ij} \geq 0$, $b_i \geq 0$, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$.
- Περιορισμός: Να ισχύουν τουλάχιστον k από τις m ανισότητες

$$\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \ i = 1, 2, \ldots, m.$$

- Εισάγουμε νέες μεταβλητές y_i, $i = 1, 2 \ldots, m$, έναν αυθαίρετα πολύ μεγάλο θετικό αριθμό M και τους περιορισμούς

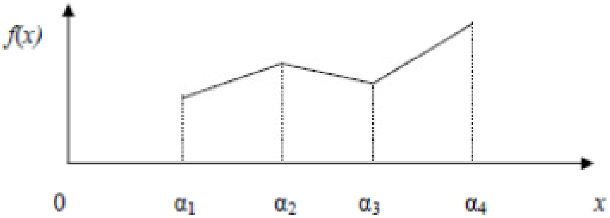
$$\begin{array}{l} \sum_{j=1}^{n} a_{ij} x_j \leq b_i + M y_i, \ i = 1, 2, \ldots, m \\ \sum_{i=1}^{m} y_i \leq m - k \\ y_i \in \{0, 1\}, \ i = 1, 2, \ldots, m. \end{array}$$

Μοντελοποίηση μεταβλ. με πεπερ. πλήθος τιμών

- Έστω μεταβλητή x με πεπερασμένο πλήθος δυνητικών τιμών από το σύνολο $\{a_1, a_2, \dots, a_m\}$
- Εισάγουμε μεταβλητές y_i, $i = 1, 2, \dots, m$ και τους περιορισμούς

$$x = \sum_{i=1}^{m} a_i y_i$$
$$\sum_{i=1}^{m} y_i = 1$$
$$y_i \in \{0, 1\},\ i = 1, 2, \dots, m.$$

Μοντελοποίηση κατά τμήματα γραμ. συναρτήσεων



- Έστω $f(x)$ συνεχής, κατά τμήματα γραμμική συνάρτηση, ορισμένη σε ένα πεπερασμένο διάστημα $[a_1, a_k]$.
- Η $f(x)$ καθορίζεται από τα σημεία $(a_i, f(a_i))$, $i = 1, 2, \ldots, k$, με $a_1 < a_2 < \cdots < a_k$.

Μοντελοποίηση κατά τμήματα γραμ. συναρτήσεων

- Κάθε $x \in [a_1, a_k]$ γράφεται ως:

$$x = \sum_{i=1}^{k} \lambda_i a_i.$$

- Η επιλογή των λ_i δεν είναι μοναδική.
- Γίνεται μοναδική αν απαιτήσουμε το πολύ δυο συνεχόμενα λ_i να είναι μη-μηδενικά. Τότε κάθε $x \in [a_i, a_{i+1}]$ γράφεται μοναδικά ως

$$x = \lambda_i a_i + \lambda_{i+1} a_{i+1},\ \lambda_i + \lambda_{i+1} = 1,\ \lambda_i, \lambda_{i+1} \geq 0.$$

- Τότε

$$f(x) = \sum_{i=1}^{k} \lambda_i f(a_i),\ x \in [a_1, a_k].$$

Μοντελοποίηση κατά τμήματα γραμ. συναρτήσεων

- Για σωστή μοντελοποίηση απαιτείται να θέσουμε τον περιορισμό:
  "Το πολύ δυο συνεχόμενα λ_i είναι μη-μηδενικά."
- Ορίζουμε $y_i \in \{0, 1\}$, $i = 1, 2, \ldots, k-1$, με

$$y_i = \begin{cases} 1 & \text{αν } a_i \leq x \leq a_{i+1}, \\ 0 & \text{διαφορετικά.} \end{cases}$$

- Επιτρέπουμε μόνο ένα y_i να είναι 1.
- Αν $y_i = 1$ τότε $\lambda_j = 0$ για $j \neq i, i+1$.

Μοντελοποίηση κατά τμήματα γραμ. συναρτήσεων

- Η βελτιστοποίηση της $f(x)$ γίνεται:

$$
\begin{array}{ll}
\min & \sum_{i=1}^{k} \lambda_i f(a_i) \\
\text{υπό} & \sum_{i=1}^{k} \lambda_i = 1, \\
& \lambda_1 \leq y_1, \\
& \lambda_i \leq y_{i-1} + y_i,\ i = 2, 3, \ldots, k-1, \\
& \lambda_k \leq y_{k-1}, \\
& \sum_{i=1}^{k-1} y_1 = 1, \\
& \lambda_i \geq 0,\ i = 1, 2, \ldots, k, \\
& y_i \in \{0, 1\},\ i = 1, 2, \ldots, k-1.
\end{array}
$$

- Αν η $f(x)$ είναι **κυρτή** κατά τμήματα γραμμική τότε υπάρχει απλούστερη μοντελοποίηση (ως π.γ.π.).

Μοντελοποίηση γραμ. συναρτήσεων με άλματα

- Σε διαδ. παραγωγής συχνά υπάρχουν κόστη εκκίνησης (setup costs).
- Σε προβλήματα μεταφοράς συχνά υπάρχουν πάγια κόστη.
- Αν x είναι η ποσότητα παραγωγής ή ή ποσότητα προς μεταφορά, το αντίστοιχο κόστος είναι συχνά

$$c(x) = \begin{cases} 0 & \text{αν } x = 0 \\ K + cx & \text{αν } x > 0. \end{cases},$$

όπου το K αφορά το πάγιο κόστος έναρξης ή μεταφοράς.

- Τέτοιες συναρτήσεις μοντελοποιούνται με χρήση δυαδικών μεταβλητών.

Μοντελοποίηση γραμ. συναρτήσεων με άλματα

- Θέτουμε

$$\begin{array}{ll} \min & Ky + cx \\ \text{υπό} & x \leq My \\ & x \geq 0 \\ & y \in \{0,1\}, \end{array}$$

όπου M κάποιος μεγάλος αριθμός (πρακτικά κάποιο άνω φράγμα που γνωρίζουμε για το x).

- Η ιδέα γενικεύεται αν έχουμε πολλές μεταβλητές και πολλά είδη κόστους εκκίνησης.

Το πρόβλημα του σακιδίου

- n αντικείμενα.
- w_j: βάρος του j.
- c_j: χρησιμότητα (αξία) του j.
- k: μέγιστο επιτρεπόμενο βάρος.
- Στόχος: Επιλ. αντικειμένων, μεγιστοπ. χρησιμότητας.
- Μεταβλητές αποφάσεων:

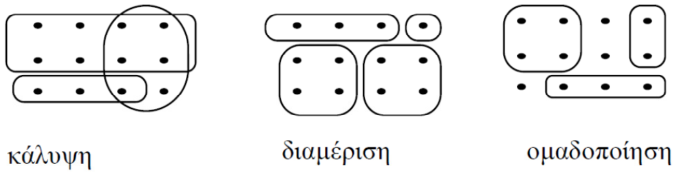
$$x_j = \begin{cases} 1, & \text{αν το } j \text{ επιλεγεί,} \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Π.α.π.:

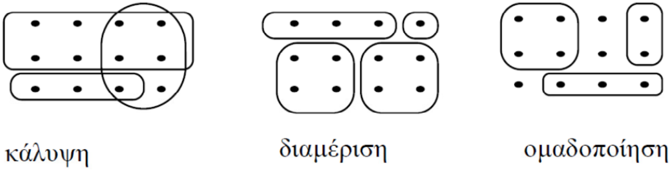
$$\begin{array}{ll} \max & \sum_{j=1}^{n} c_j x_j \\ \text{υπό} & \sum_{j=1}^{n} w_j x_j \le k \\ & x_j \in \{0,1\},\ j = 1,2,\ldots,n. \end{array}$$

Τα προβλήματα κάλυψης, ομαδοπ., διαμέρισης

- $M = \{1, 2, \ldots, m\}$: Σύνολο σημείων.
- $N = \{1, 2, \ldots, n\}$: Σύνολο δεικτών.
- $\{M_1, M_2, \ldots, M_n\}$: Συλλογή υποσυνόλων του M.
- c_j: Βάρος συνόλου M_j.
- $F \subseteq N$ κάλυψη του $M \Leftrightarrow \cup_{j \in F} M_j = M$.
- $F \subseteq N$ ομαδοποίηση του $M \Leftrightarrow M_j \cap M_k = \emptyset$, για $j, k \in F$, $j \neq k$.
- $F \subseteq N$ διαμέριση του $M \Leftrightarrow F$ κάλυψη και ομαδοποίηση του M.



Τα προβλήματα κάλυψης, ομαδοπ., διαμέρισης



- Πρόβλημα βέλτιστης κάλυψης: Κάλυψη ελάχ. βάρους.
- Πρόβλημα βέλτιστης ομαδοπ.: Ομαδοπ. μέγ. βάρους.
- Πρόβλημα βέλτιστης διαμερ.: Διαμέρ. ελαχ./μεγ. βάρους (2 εκδοχές).

Τα προβλήματα κάλυψης, ομαδοπ., διαμέρισης

- $M = \{1, 2, \ldots, m\}$: Σύνολο σημείων.
- $\{M_1, M_2, \ldots, M_n\}$: Συλλογή υποσυνόλων του M.
- $A = (a_{ij})$ $m \times n$ πίνακας: Πίνακας πρόσπτωσης της συλλογής $\{M_1, M_2, \ldots, M_n\}$:

$$a_{ij} = \begin{cases} 1, & \text{αν } i \in M_j \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Μεταβλητές αποφάσεων:

$$x_j = \begin{cases} 1, & \text{αν } j \in F \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Διάνυσμα αποφάσεων $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$.

Τα προβλήματα κάλυψης, ομαδοπ., διαμέρισης

- $A = (a_{ij})$ $m \times n$ πίνακας πρόσπτωσης:

$$a_{ij} = \begin{cases} 1, & \text{αν } i \in M_j \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Μεταβλητές αποφάσεων:

$$x_j = \begin{cases} 1, & \text{αν } j \in F \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Διάνυσμα αποφάσεων $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$.
  $\mathbf{e} = (1, 1, \ldots, 1)^T$.
- F κάλυψη $\Leftrightarrow A\mathbf{x} \geq \mathbf{e}$.
  F ομαδοποίηση $\Leftrightarrow A\mathbf{x} \leq \mathbf{e}$.
  F διαμέριση $\Leftrightarrow A\mathbf{x} = \mathbf{e}$.

Τα προβλήματα κάλυψης, ομαδοπ., διαμέρισης

- Π.α.π. κάλυψης, ομαδοποίησης, διαμέρισης:

$$
\begin{array}{ll}
\max & \sum_{j=1}^{n} c_j x_j \\
\text{υπό} & \sum_{j=1}^{n} a_{ij} x_j \geq (\leq, =) 1, \ i = 1, 2, \ldots m, \\
& x_j \in \{0, 1\}, \ j = 1, 2, \ldots, n.
\end{array}
$$

Το πρόβλημα σχεδιασμού πτήσεων πληρωμάτων αεροπορικής εταιρείας - Crew scheduling problem

- Αεροπορική εταιρεία πρέπει να καλύψει τις 11 πτήσεις που έχει αναλάβει ανά ημέρα με 3 πληρώματα.
- Υπάρχουν 12 δυνατές ακολουθίες πτήσεων που σέβονται
    - τη χρονική αλληλουχία των πτήσεων,
    - τα ελάχιστα χρονικά διαστήματα μεταξύ των πτήσεων,
    - τους κανονισμούς προσωπικού
- Κάθε ακολουθία πτήσεων περιγράφει ποιές πτήσεις θα κάνει ένα πλήρωμα και με τί σειρά και το αντίστοιχο κόστος.
- Υπάρχει η δυνατότητα να μεταφερθεί περισσότερο από ένα πλήρωμα σε μια πτήση.
- Πρέπει να καλυφθούν όλες οι πτήσεις με πληρώματα με το ελάχιστο κόστος.

Το πρόβλημα σχεδιασμού πτήσεων πληρωμάτων αεροπορικής εταιρείας - Crew scheduling problem

Πτήση	1	2	3	4	5	6	7	8	9	10	11	12
Αθ-Θεσ	1			1			1			1		
Αθ-Ηρακ		1			1			1			1	
Αθ-Ροδ			1			1			1			1
Θεσ-Ιωαν				2			2		3	2		3
Θεσ-Αθ	2					3				5	5	
Ιωαν-Ηρακ				3	3				4			
Ιωαν-Ροδ							3	3		3	3	4
Ηρακ-Αθ		2		4	4				5			
Ηρακ-Ιωαν					2			2			2	
Ροδ-Αθ			2				4	4				5
Ροδ-Θεσ						2			2	4	4	2
Κόστος (σε χιλ.)	2	3	4	6	7	5	7	8	9	9	8	9

Το πρόβλημα σχεδιασμού πτήσεων πληρωμάτων αεροπορικής εταιρείας - Crew scheduling problem

- Πρόκειται για πρόβλημα κάλυψης.
- Το σύνολο των σημείων M είναι το σύνολο των πτήσεων.
- Η συλλογή των υποσυνόλων είναι η συλλογή των ακολουθιών πτήσεων.
- Ο πίνακας πρόσπτωσης είναι ο $A = (a_{ij})$ με

$$a_{ij} = \begin{cases} 1 & \text{αν η πτήση } i \text{ ανήκει στην ακολουθία } j \\ 0 & \text{διαφορετικά.} \end{cases}$$

- Μεταβλητές αποφάσεων

$$x_j = \begin{cases} 1 & \text{αν η ακολουθία } j \text{ επιλεγεί} \\ 0 & \text{διαφορετικά.} \end{cases}$$

Το πρόβλημα σχεδιασμού πτήσεων πληρωμάτων αεροπορικής εταιρείας - Crew scheduling problem

- Η βέλτιστη λύση είναι $x_3 = x_4 = x_{11} = 1$ και τα υπόλοιπα 0.
- Δηλαδή 3 πληρώματα για
  1. Αθήνα - Ρόδο - Αθήνα,
  2. Αθήνα - Θεσσαλονίκη - Ιωάννινα - Ηράκλειο - Αθήνα,
  3. Αθήνα - Ηράκλειο - Ιωάννινα - Ρόδο - Θεσσαλονίκη - Αθήνα.
- Βέλτιστο κόστος: 18000 ευρώ.
- Σχετικό είναι και το πρόβλημα του σχεδιασμού πτήσεων αεροσκαφών αεροπορικής εταιρείας.
- Αυτό αντιστοιχεί σε πρόβλημα διαμέρισης (αφού δεν ταξιδεύει κενό αεροσκάφος σε κάποια διαδρομή).

Το πρόβλημα της ανάθεσης

- m άτομα: $A_1, A_2, \ldots, A_m$, m εργασίες: $B_1, B_2, \ldots, B_m$.
- Ακριβώς ένα άτομο για κάθε εργασία.
- r_{ij}: Απόδοση του ατόμου A_i στην εργασία B_j.
- Στόχος: Ανάθεση εργασιών για μέγιστη απόδοση.
- Μεταβλητές αποφάσεων:

$$x_{ij} = \begin{cases} 1, & \text{αν το άτομο } A_i \text{ ασχοληθεί με την } B_j \\ 0, & \text{διαφορετικά.} \end{cases}$$

- Π.α.π.:

$$\begin{array}{ll} \max & \sum_{i=1}^{m} \sum_{j=1}^{m} r_{ij} x_{ij} \\ \text{υπό} & \sum_{j=1}^{m} x_{ij} = 1,\ i = 1, 2, \ldots, m \\ & \sum_{i=1}^{m} x_{ij} = 1,\ j = 1, 2, \ldots, m \\ & x_{ij} \in \{0, 1\},\ i, j = 1, 2, \ldots, n. \end{array}$$

Το πρόβλημα επιλογής τοποθεσίας εγκαταστάσεων

- n δυνατές τοποθεσίες παροχής υπηρεσιών.
- m πελάτες.
- c_j: Κόστος εγκατάστασης παρόχου υπηρεσιών στην τοποθεσία j.
- d_{ij}: Κόστος ανάθεσης πελάτη i στον πάροχο j.
- Στόχος: Επιλογή τοποθεσιών για εγκατάσταση παρόχων υπηρεσιών και ανάθεση κάθε πελάτη σε έναν πάροχο, με ελάχιστο δυνατό κόστος.
- Μεταβλητές αποφάσεων:

$$y_j = \begin{cases} 1 & \text{αν η } j \text{ τοποθεσία χρησιμοποιηθεί για πάροχο} \\ 0 & \text{διαφορετικά} \end{cases}$$

$$x_{ij} = \begin{cases} 1 & \text{αν ο πελάτης } i \text{ ανατεθεί στον πάροχο } j \\ 0 & \text{διαφορετικά.} \end{cases}$$

Το πρόβλημα επιλογής τοποθεσίας εγκαταστάσεων

- Η κλασική μοντελοποίηση Facility Location (FL) - Π.α.π.:

$$
\begin{array}{ll}
\min & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
\text{υπό} & \sum_{j=1}^{n} x_{ij} = 1, \; i = 1, 2, \ldots, m \\
 & x_{ij} \leq y_j, \; i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n \\
 & x_{ij} \in \{0, 1\}, \; i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n \\
 & y_j \in \{0, 1\}, \; j = 1, 2, \ldots, n.
\end{array}
$$

Το πρόβλημα της επιλογής τοποθεσίας εγκαταστάσεων

- Η εναλλακτική μοντελοποίηση Aggregate Facility Location (AFL) - Π.α.π.:

$$
\begin{array}{ll}
\min & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
\text{υπό} & \sum_{j=1}^{n} x_{ij} = 1,\ i = 1, 2, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} \leq m y_j,\ j = 1, 2, \ldots, n \\
& x_{ij} \in \{0, 1\},\ i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n \\
& y_j \in \{0, 1\},\ j = 1, 2, \ldots, n.
\end{array}
$$

Προγραμματισμός βαρδιών

- Τηλεφωνικό κέντρο θέλει να προγραμματίσει τις βάρδιες προσωπικού σε εβδομαδιαία βάση.
- d_j: Ο αριθμός των απαιτούμενων υπαλλήλων τη μέρα j, $j = 1, 2, \ldots, 7$.
- Κάθε υπάλληλος δουλεύει 5 συνεχόμενες μέρες και μετά έχει 2 συνεχόμενες μέρες ρεπό.
- Στόχος είναι να βρεθεί ο προγραμματισμός των βαρδιών του προσωπικού που ελαχιστοποιεί τον αριθμό των απαιτούμενων υπαλλήλων σε εβδομαδιαία βάση.

Προγραμματισμός βαρδιών - Μοντελοποίηση

- x_j: αριθμός υπαλλήλων που αρχίζουν να δουλεύουν τη μέρα j.
- Π.α.π.:

$$
\begin{array}{llllllllll}
\min & \sum_{j=1}^{7} x_j & & & & & & & & \\
\text{υπό} & x_1 & & & +x_4 & +x_5 & +x_6 & +x_7 & \geq & d_1 \\
& x_1 & +x_2 & & & +x_5 & +x_6 & +x_7 & \geq & d_2 \\
& x_1 & +x_2 & +x_3 & & & +x_6 & +x_7 & \geq & d_3 \\
& x_1 & +x_2 & +x_3 & +x_4 & & & +x_7 & \geq & d_4 \\
& x_1 & +x_2 & +x_3 & +x_4 & +x_5 & & & \geq & d_5 \\
& & x_2 & +x_3 & +x_4 & +x_5 & +x_6 & & \geq & d_6 \\
& & & x_3 & +x_4 & +x_5 & +x_6 & +x_7 & \geq & d_7
\end{array}
$$

$$x_j \in \{0, 1, 2, \ldots\}.$$

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Γενίκευση πρόβληματος επιλογής τοποθεσίας εγκαταστάσεων - Δεδομένα:
  1. n δυνατές τοποθεσίες παροχής υπηρεσιών.
  2. m πελάτες.
  3. c_j κόστος εγκατάστασης στην j.
  4. d_{ij} κόστος ανάθεσης του i στην j.
- Μεταβλητές αποφάσεων:

$$\begin{aligned} y_j &= \begin{cases} 1 & \text{αν η } j \text{ τοποθεσία χρησιμοποιηθεί για πάροχο} \\ 0 & \text{διαφορετικά} \end{cases} \\ x_{ij} &= \text{ποσοστό ζήτησης του } i \text{ που ικανοποιείται από τη } j \end{aligned}$$

- Υπάρχουν εναλλακτικές μοντελοποιήσεις.

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Η κλασική μοντελοποίηση Facility Location (FL) - Π.α.π.:

$$
\begin{array}{ll}
\min & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
\text{υπό} & \sum_{j=1}^{n} x_{ij} = 1, \; i = 1, 2, \ldots, m \\
& x_{ij} \leq y_j, \; i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n \\
& 0 \leq x_{ij} \leq 1, \; i = 1, 2, \ldots, m, \; j = 1, 2, \ldots, n \\
& y_j \in \{0, 1\}, \; j = 1, 2, \ldots, n.
\end{array}
$$

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Η εναλλακτική μοντελοποίηση Aggregate Facility Location (AFL) - Π.α.π.:

$$
\begin{array}{ll}
\min & \sum_{j=1}^{n} c_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} x_{ij} \\
\text{υπό} & \sum_{j=1}^{n} x_{ij} = 1,\ i = 1, 2, \ldots, m \\
& \sum_{i=1}^{m} x_{ij} \leq m y_j,\ j = 1, 2, \ldots, n \\
& 0 \leq x_{ij} \leq 1,\ i = 1, 2, \ldots, m,\ j = 1, 2, \ldots, n \\
& y_j \in \{0, 1\},\ j = 1, 2, \ldots, n.
\end{array}
$$

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Ποιά από τις δυο μοντελοποιήσεις είναι προτιμότερη;

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Οι εφικτές περιοχές και των δυο μοντελοποιήσεων είναι οι ίδιες.
- Η βέλτιστη τιμή της αντικειμενικής συνάρτησης είναι η ίδια και για τις δυο μοντελοποιήσεις.
- Οι εφικτές περιοχές των χαλαρώσεων γραμμικού προγραμματισμού (χ.γ.π.) δεν είναι οι ίδιες.
- Έστω
    1. P_{FL} : Εφικτή περιοχή της χ.γ.π. της FL,
    2. P_{AFL} : Εφικτή περιοχή της χ.γ.π. της AFL,
    3. I_{FL} : Εφικτή περιοχή της FL,
    4. I_{AFL} : Εφικτή περιοχή της AFL.
- Είναι $I_{AFL} = I_{FL} \subseteq P_{FL} \subseteq P_{AFL}$.

Επιλογή καλής μοντελοποίησης - Παράδειγμα

- Η βελτιστοποίηση πάνω στις I_{AFL}, I_{FL} είναι δύσκολη.
- Οι εφικτές περιοχές π.γ.π. είναι μη συνεκτικά, μη-κυρτά σύνολα.
- Η βελτιστοποίηση πάνω στις P_{AFL}, P_{FL} είναι πολύ ευκολότερη. Έχουμε π.γ.π.
- Αν τύχει η βέλτιστη λύση της χ.γ.π. του π.α.π. να ικανοποιεί τους περιορισμούς ακεραιότητας, τότε είναι και βέλτιστη λύση του π.α.π.
- Άρα καλύτερη μοντελοποίηση είναι αυτή που δίνει μικρότερη εφικτή περιοχή της χ.γ.π.
- Στο παράδειγμα η FL είναι καλύτερη μοντελοποίηση, παρότι έχει περισσότερους περιορισμούς από την AFL.
- Μια μοντελοποίηση ακέραιου προγραμματισμού είναι καλύτερη μιας άλλης αν η εφικτή περιοχή της χ.γ.π. της είναι μικρότερη της εφικτής περιοχής της χ.γ.π. της