

Προηγμένα Θέματα Αλγορίθμων

Γραμμικός Προγραμματισμός II

Βασίλειος Νάκος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών
Πανεπιστήμιο Αθηνών

Πρωτεύον

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Δυϊκό

$$\begin{aligned} \text{maximize} \quad & b^\top y \\ & A^\top y \leq c \\ & y \geq 0 \end{aligned}$$

Πρωτεύον

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Δυϊκό

$$\begin{aligned} \text{maximize} \quad & b^\top y \\ & A^\top y \leq c \\ & y \geq 0 \end{aligned}$$

Επιλυσιμότητα Γραμμικού Προγραμματισμού.

Ένα Γραμμικό Πρόγραμμα μπορεί να λυθεί σε πολυωνυμικό χρόνο ως προς το n (πλήθος μεταβλητών), το m (πλήθος περιορισμών) και το μήκος των αριθμών στην είσοδο (στα A, b, c).

Τι δεν μπορεί να λυθεί σε πολυωνμικό χρόνο;

★ Ακέραιος Προγραμματισμός.

Πρωτεύον

minimize $c^\top x$

$Ax \geq b$

$x \in \{0, 1\}^n$

Δυϊκό

maximize $b^\top y$

$A^\top y \leq c$

$y \in \{0, 1\}^m$

Δυσκολία Ακέραιου Προγραμματισμού.

Ο Ακέραιος Προγραμματισμός είναι NP-δύσκολος.

Απόδειξη NP-πληρότητας.

Έστω μία 3-SAT φόρμουλα με n μεταβλητές και m περιορισμούς.

$$\phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_3 \vee \neg x_4)$$

Απόδειξη NP-πληρότητας.

Έστω μία 3-SAT φόρμουλα με n μεταβλητές και m περιορισμούς.

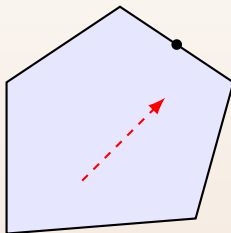
$$\phi = (x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge (x_2 \vee x_4 \vee \neg x_5) \wedge (x_1 \vee \neg x_3 \vee \neg x_4)$$

Για κάθε μεταβλητή φτιάχνουμε μία ακέραια μεταβλητή $x_i \in \{0, 1\}$ και κάθε φράση αντιστοιχεί σε έναν περιορισμό.

$$\begin{aligned} \text{minimize} \quad & \sum_i 0 \\ & x_1 + (1 - x_2) + x_2 \geq 1 \\ & x_1 + x_3 + (1 - x_4) \geq 1, \\ & x_2 + x_4 + (1 - x_5) \geq 1, \\ & x_1 + x_3 + (1 - x_4) \geq 1, \\ & x_1, x_2, x_3, x_4, x_5 \in \{0, 1\} \end{aligned}$$

Τι είναι αυτό που κάνει τον Γραμμικό Προγραμματισμό πιο εύκολο;

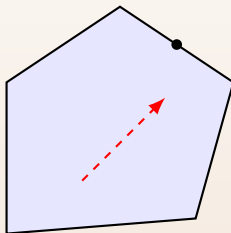
👁️ Θυμάστε που είχαμε πει πως η βέλτιστη λύση (εφόσον το πολύεδρο είναι μη κενό και πεπερασμένου κέρδους) μπορεί να βρεθεί σε **κορυφή** του πολυέδρου;



Αυτό δεν ισχύει στον Ακέραιο Προγραμματισμό.

Τι είναι αυτό που κάνει τον Γραμμικό Προγραμματισμό πιο εύκολο;

👁️ Θυμάστε που είχαμε πει πως η βέλτιστη λύση (εφόσον το πολύεδρο είναι μη κενό και πεπερασμένου κέρδους) μπορεί να βρεθεί σε **κορυφή** του πολυέδρου;



Αυτό δεν ισχύει στον Ακέραιο Προγραμματισμό.
Πάμε να το δούμε τι σημαίνει κορυφή στην γενική περίπτωση.

Κανονική Μορφή II.

☛ Κάθε Γραμμικό Πρόγραμμα μπορεί να γραφτεί είτε στην Κανονική Μορφή I είτε στην Κανονική Μορφή II.

Κανονική Μορφή I

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Κανονική Μορφή II.

☛ Κάθε Γραμμικό Πρόγραμμα μπορεί να γραφτεί είτε στην Κανονική Μορφή I είτε στην Κανονική Μορφή II.

Κανονική Μορφή I

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax \geq b \\ & x \geq 0 \end{aligned}$$

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Θα μελετήσουμε την Κανονική Μορφή II.

👉 Μπορούμε να πετάξουμε γραμμές του A ώστε να μείνουμε μόνο με γραμμικώς ανεξάρτητες γραμμές.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

👉 Μπορούμε να πετάξουμε γραμμές του A ώστε να μείνουμε μόνο με γραμμικώς ανεξάρτητες γραμμές.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Άρα, αν μείνουμε με m γραμμές, ο βαθμός του A είναι m και έχει και ακριβώς m γραμμικώς ανεξάρτητες στήλες.

Έστω $B \subseteq [n], |B| = m$ (υποσύνολο στηλών), έστω $N := [n] \setminus B$.

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & A_B x_B + A_N x_N = b \\ & x \geq 0 \end{aligned}$$

Έστω $B \subseteq [n], |B| = m$ (υποσύνολο στηλών), έστω $N := [n] \setminus B$.

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & A_B x_B + A_N x_N = b \\ & x \geq 0 \end{aligned}$$

👉 Αν οι στήλες αυτές είναι γραμμικώς ανεξάρτητες οι μεταβλητές $\{x_i : i \in B\}$ είναι βασικές και οι υπόλοιπες ελεύθερες

Έστω $B \subseteq [n], |B| = m$ (υποσύνολο στηλών), έστω $N := [n] \setminus B$.

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & A_B x_B + A_N x_N = b \\ & x \geq 0 \end{aligned}$$

👉 Αν οι στήλες αυτές είναι γραμμικώς ανεξάρτητες οι μεταβλητές $\{x_i : i \in B\}$ είναι βασικές και οι υπόλοιπες ελεύθερες $\Rightarrow$ μπορώ να θέσω x_N ό,τι θέλω και υπάρχει πάντα μοναδική λύση.

Έστω $B \subseteq [n], |B| = m$ (υποσύνολο στηλών), έστω $N := [n] \setminus B$.

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & A_B x_B + A_N x_N = b \\ & x \geq 0 \end{aligned}$$

☞ Αν οι στήλες αυτές είναι γραμμικώς ανεξάρτητες οι μεταβλητές $\{x_i : i \in B\}$ είναι βασικές και οι υπόλοιπες ελεύθερες $\Rightarrow$ μπορώ να θέσω x_N ό,τι θέλω και υπάρχει πάντα μοναδική λύση.

★ Θέτω $x_N = 0$ και τότε παίρνω μοναδικό x_B .

Κορυφές του Πολυέδρου.

Έστω $B \subseteq [n], |B| = m$ (υποσύνολο στηλών), έστω $N := [n] \setminus B$.

Κανονική Μορφή II

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & A_B x_B + A_N x_N = b \\ & x \geq 0 \end{aligned}$$

☛ Αν οι στήλες αυτές είναι γραμμικώς ανεξάρτητες οι μεταβλητές $\{x_i : i \in B\}$ είναι βασικές και οι υπόλοιπες ελεύθερες $\Rightarrow$ μπορώ να θέσω $x_N = 0$, τι θέλω και υπάρχει πάντα μοναδική λύση.

★ Θέτω $x_N = 0$ και τότε παίρνω μοναδικό x_B . Αν $x_B \geq 0$ οπρίζω ως κορυφή του πολυέδρου το x που ταυτίζεται με το x_B στα $i \in B$ και είναι 0 για $i \notin B$. Άρα υπάρχουν $\binom{n}{m}$ πιθανές κορυφές.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Μία βέλτιστη λύση του Γραμμικού Προγράμματος προκύπτει πάντα σε κορυφή του πολυέδρου.

Κορυφές του Πολυέδρου.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Μία βέλτιστη λύση του Γραμμικού Προγράμματος προκύπτει πάντα σε κορυφή του πολυέδρου.

👉 Κοινώς, υπάρχει υποσύνολο γραμμικώς ανεξάρτητων στηλών B μεγέθους m ώστε μία βέλτιστη λύση του Γραμμικού Προγράμματος να ικανοποιεί $x_i = 0$ για κάθε $i \notin B$.

$$\begin{aligned} \text{minimize} \quad & c^\top x \\ & Ax = b \\ & x \geq 0 \end{aligned}$$

Μία βέλτιστη λύση του Γραμμικού Προγράμματος προκύπτει πάντα σε κορυφή του πολυέδρου.

👉 Κοινώς, υπάρχει υποσύνολο γραμμικώς ανεξάρτητων στηλών B μεγέθους m ώστε μία βέλτιστη λύση του Γραμμικού Προγράμματος να ικανοποιεί $x_i = 0$ για κάθε $i \notin B$.

Αυτή η ιδιότητα δεν ισχύει στον Ακέραιο Προγραμματισμό!

Βασική γεωμετρική εικόνα

- 1 Η μέθοδος Simplex ξεκινά από μία *κορυφή* του εφικτού πολυέδρου.

Βασική γεωμετρική εικόνα

- 1 Η μέθοδος Simplex ξεκινά από μία κορυφή του εφικτού πολυέδρου.
- 2 Σε κάθε βήμα κινείται σε μία γειτονική κορυφή με μικρότερο κόστος (ή μεγαλύτερο κέρδος), δηλαδή σε ένα νέο σύνολο στηλών B' που διαφέρει ακριβώς σε μία στήλη από το τρέχον σύνολο B .

Βασική γεωμετρική εικόνα

- 1 Η μέθοδος Simplex ξεκινά από μία κορυφή του εφικτού πολυέδρου.
- 2 Σε κάθε βήμα κινείται σε μία γειτονική κορυφή με μικρότερο κόστος (ή μεγαλύτερο κέρδος), δηλαδή σε ένα νέο σύνολο στηλών B' που διαφέρει ακριβώς σε μία στήλη από το τρέχον σύνολο B .
- 3 Ο αλγόριθμος σταματά όταν δεν υπάρχει γειτονική κορυφή που μειώνει την αντικειμενική συνάρτηση (όχι ακριβώς).

Βασική γεωμετρική εικόνα

- 1 Η μέθοδος Simplex ξεκινά από μία κορυφή του εφικτού πολυέδρου.
- 2 Σε κάθε βήμα κινείται σε μία γειτονική κορυφή με μικρότερο κόστος (ή μεγαλύτερο κέρδος), δηλαδή σε ένα νέο σύνολο στηλών B' που διαφέρει ακριβώς σε μία στήλη από το τρέχον σύνολο B .
- 3 Ο αλγόριθμος σταματά όταν δεν υπάρχει γειτονική κορυφή που μειώνει την αντικειμενική συνάρτηση (όχι ακριβώς).
- 4 Αποδεικνύεται ότι τότε βρισκόμαστε σε βέλτιστη κορυφή.

Πλεονεκτήματα

- Πολύ καθαρή γεωμετρική διαισθητική εικόνα (κίνηση από κορυφή σε κορυφή).

Μειονεκτήματα

Πλεονεκτήματα

- Πολύ καθαρή γεωμετρική διαισθητική εικόνα (κίνηση από κορυφή σε κορυφή).
- Εξαιρετικά αποδοτικός στην πράξη για μεγάλο εύρος πραγματικών προβλημάτων.

Μειονεκτήματα

Πλεονεκτήματα

- Πολύ καθαρή γεωμετρική διαισθητική εικόνα (κίνηση από κορυφή σε κορυφή).
- Εξαιρετικά αποδοτικός στην πράξη για μεγάλο εύρος πραγματικών προβλημάτων.
- Δίνει απευθείας *βασική βέλτιστη λύση* και πληροφoρία για το δυϊκό πρόβλημα.

Μειονεκτήματα

Πλεονεκτήματα

- Πολύ καθαρή γεωμετρική διαισθητική εικόνα (κίνηση από κορυφή σε κορυφή).
- Εξαιρετικά αποδοτικός στην πράξη για μεγάλο εύρος πραγματικών προβλημάτων.
- Δίνει απευθείας *βασική βέλτιστη λύση* και πληροφορία για το δυϊκό πρόβλημα.

Μειονεκτήματα

- Εκθετικός χρόνος στη χειρότερη περίπτωση (θεωρητικά).

Πλεονεκτήματα

- Πολύ καθαρή γεωμετρική διαισθητική εικόνα (κίνηση από κορυφή σε κορυφή).
- Εξαιρετικά αποδοτικός στην πράξη για μεγάλο εύρος πραγματικών προβλημάτων.
- Δίνει απευθείας βασική βέλτιστη λύση και πληροφορία για το δυϊκό πρόβλημα.

Μειονεκτήματα

- Εκθετικός χρόνος στη χειρότερη περίπτωση (θεωρητικά).
- Πιθανή εμφάνιση κύκλου – χρειάζονται ειδικοί κανόνες (π.χ. Bland) για αποφυγή.

Κάλυμμα Συνόλου και Ακέραιος Προγραμματισμός.

★ Δοθέντος γράφου $G(V, E)$ να βρεθεί ένα σύνολο κορυφών $V^* \subseteq V$ ελάχιστης πληθικότητας έτσι ώστε καθε ακμή $e \in E$ έχει τουλάχιστον ένα άκρο της στο V^* .

Κάλυμμα Συνόλου και Ακέραιος Προγραμματισμός.

★ Δοθέντος γράφου $G(V, E)$ να βρεθεί ένα σύνολο κορυφών $V^* \subseteq V$ ελάχιστης πληθικότητας έτσι ώστε καθε ακμή $e \in E$ έχει τουλάχιστον ένα άκρο της στο V^* .

Κάλυμμα Συνόλου και Ακέραιος Προγραμματισμός.

★ Δοθέντος γράφου $G(V, E)$ να βρεθεί ένα σύνολο κορυφών $V^* \subseteq V$ ελάχιστης πληθικότητας έτσι ώστε καθε ακμή $e \in E$ έχει τουλάχιστον ένα άκρο της στο V^* .

$$\begin{aligned} \text{minimize } & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & x_u \in \{0, 1\}, \forall u \in V \end{aligned}$$

Κάλυμμα Συνόλου και Ακέραιος Προγραμματισμός.

★ Δοθέντος γράφου $G(V, E)$ να βρεθεί ένα σύνολο κορυφών $V^* \subseteq V$ ελάχιστης πληθικότητας έτσι ώστε καθε ακμή $e \in E$ έχει τουλάχιστον ένα άκρο της στο V^* .

$$\begin{aligned} \text{minimize } & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & x_u \in \{0, 1\}, \forall u \in V \end{aligned}$$

Δεν μπορούμε να το λύσουμε γρήγορα, αλλά μπορούμε να λύσουμε το αντίστοιχο **γραμμικό πρόγραμμα**.

Μπορούμε να κάνουμε κάτι με εκείνη τη λύση;

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} &\text{minimize } \sum_{u \in V} x_u \\ &x_u + x_v \geq 1, \forall e = (u, v) \in E \\ &\textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} \text{minimize } & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} \text{minimize } & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Γιατί αυτό είναι ένα έγκυρο κάλυμμα;

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} &\text{minimize } \sum_{u \in V} x_u \\ &x_u + x_v \geq 1, \forall e = (u, v) \in E \\ &\textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Γιατί αυτό είναι ένα έγκυρο κάλυμμα;

Κόστος καλύμματος

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} \text{minimize} \quad & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Γιατί αυτό είναι ένα έγκυρο κάλυμμα;

$$\text{Κόστος καλύμματος} = |S| \leq 2 \cdot \sum_{u \in V} x_u$$

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} & \text{minimize} \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Γιατί αυτό είναι ένα έγκυρο κάλυμμα;

$$\text{Κόστος καλύμματος} = |S| \leq 2 \cdot \sum_{u \in V} x_u \leq$$

Στρογγυλοποιώντας την Κλασματική Λύση.

Έστω x^* μία βέλτιστη λύση στο 'χαλαρωμένο' γραμμικό πρόγραμμα.

$$\begin{aligned} \text{minimize} \quad & \sum_{u \in V} x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Έστω το σύνολο κορυφών $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ ως πιθανό κάλυμμα.

Γιατί αυτό είναι ένα έγκυρο κάλυμμα;

Κόστος καλύμματος $= |S| \leq 2 \cdot \sum_{u \in V} x_u \leq$
 $2 \cdot (\text{Κόστος καλύτερης ακέραιας λύσης})$ (γιατί;).

Βεβαρυμένο σύνολο κορυφών.

Μπορούμε να κάνουμε το ίδιο όταν οι κορυφές έχουν βάρη c_u .

$$\text{minimize } \sum_{u \in V} c_u x_u$$

$$x_u + x_v \geq 1, \forall e = (u, v) \in E$$

$$x_u \geq 0, \forall u \in V$$

Βεβαρυμένο σύνολο κορυφών.

Μπορούμε να κάνουμε το ίδιο όταν οι κορυφές έχουν βάρη c_u .

$$\begin{aligned} &\text{minimize } \sum_{u \in V} c_u x_u \\ &x_u + x_v \geq 1, \forall e = (u, v) \in E \\ &\textcolor{red}{x_u} \geq 0, \forall u \in V \end{aligned}$$

Ορίζουμε $S := \{u \mid x_u^* \geq \frac{1}{2}\}$

Βεβαρυμένο σύνολο κορυφών.

Μπορούμε να κάνουμε το ίδιο όταν οι κορυφές έχουν βάρη c_u .

$$\begin{aligned} &\text{minimize } \sum_{u \in V} c_u x_u \\ &x_u + x_v \geq 1, \forall e = (u, v) \in E \\ &\textcolor{red}{x_u} \geq \textcolor{red}{0}, \forall u \in V \end{aligned}$$

Ορίζουμε $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ και παίρνουμε έγκυρο ακέραιο κάλυμμα κορυφών το πολύ διπλασίου κόστους από την βέλτιστη ακέραια λύση.

Βεβαρυμένο σύνολο κορυφών.

Μπορούμε να κάνουμε το ίδιο όταν οι κορυφές έχουν βάρη c_u .

$$\begin{aligned} & \text{minimize} \sum_{u \in V} c_u x_u \\ & x_u + x_v \geq 1, \forall e = (u, v) \in E \\ & \textcolor{red}{x_u} \geq 0, \forall u \in V \end{aligned}$$

Ορίζουμε $S := \{u \mid x_u^* \geq \frac{1}{2}\}$ και παίρνουμε έγκυρο ακέραιο κάλυμμα κορυφών το πολύ διπλασίου κόστους από την βέλτιστη ακέραια λύση.

👉 Συχνά λύνουμε το γραμμικό πρόγραμμα και στρογγυλοποιούμε την λύση για να πάρουμε μία *ακέραια*.

$$\text{minimize } \sum_{u \in V} c_u x_u$$

$$x_u + x_v \geq 1, \forall e = (u, v) \in E$$

$$x_u \geq 0, \forall u \in V$$

Ποιό είναι το δυϊκό;

$$\text{minimize } \sum_{u \in V} c_u x_u$$

$$x_u + x_v \geq 1, \forall e = (u, v) \in E$$

$$x_u \geq 0, \forall u \in V$$

$$\text{maximize } \sum_{e \in E} y_e$$

$$\sum_{e: u \in e} y_e \leq c_u, \forall u \in V$$

$$y_e \geq 0, \forall e \in E$$

Πρόβλημα Χωροθέτησης Υπηρεσιών

Δίνονται δύο σύνολα C, F σε ένα χώρο με μετρική την απόσταση d . Σε κάθε θέση του C υπάρχει ένας πελάτης. Σε κάθε $i \in F$ μπορούμε να ανοίξουμε μια υπηρεσία με κόστος f_i .

Πρόβλημα Χωροθέτησης Υπηρεσιών

Δίνονται δύο σύνολα C, F σε ένα χώρο με μετρική την απόσταση d . Σε κάθε θέση του C υπάρχει ένας πελάτης. Σε κάθε $i \in F$ μπορούμε να ανοίξουμε μια υπηρεσία με κόστος f_i .

★ Θέλουμε να ανοίξουμε κάποιες υπηρεσίες $F' \subseteq F$ και να αναθέσουμε κάθε πελάτη $j \in C$ σε κάποια υπηρεσία $\sigma(j) \in F'$: το κόστος είναι $d_{\sigma(j),j}$ για την ανάθεση.

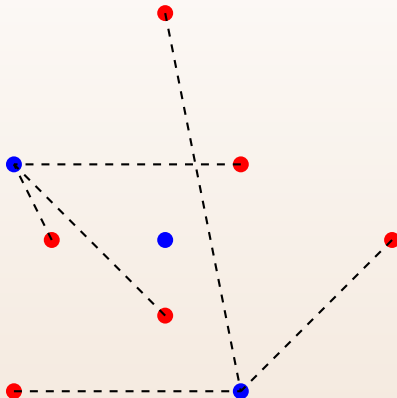
Πρόβλημα Χωροθέτησης Υπηρεσιών

Δίνονται δύο σύνολα C, F σε ένα χώρο με μετρική την απόσταση d . Σε κάθε θέση του C υπάρχει ένας πελάτης. Σε κάθε $i \in F$ μπορούμε να ανοίξουμε μια υπηρεσία με κόστος f_i .

★ Θέλουμε να ανοίξουμε κάποιες υπηρεσίες $F' \subseteq F$ και να αναθέσουμε κάθε πελάτη $j \in C$ σε κάποια υπηρεσία $\sigma(j) \in F'$: το κόστος είναι $d_{\sigma(j),j}$ για την ανάθεση.

$$C_{ALG} = \underbrace{\sum_{j \in F'} f_i}_{\text{κόστος ανοίγματος}} + \underbrace{\sum_{i \in C} d_{\sigma(j),j}}_{\text{κόστος ανάθεσης}}$$

Στο παρακάτω στιγμιότυπο τα κόκκινα σημεία αντιστοιχούν στους πελάτες και τα μπλε στις πιθανές υπηρεσίες. Είναι ανοικτές ακριβώς δύο υπηρεσίες.



Πρωτεύον και Δυϊκό για Πρόβλημα Χωροθέτησης Υπηρεσιών

Πρωτεύον

$$\text{minimize } \sum_{i \in F} y_i + \sum_{i \in F, j \in C} d_{i,j} x_{i,j}$$

$$\sum_{i \in F} x_{ij} \geq 1, \forall j \in C$$

$$y_i \geq x_{ij}, \forall i \in F, j \in C$$

$$x_{ij}, y_i \geq 0$$

Δυϊκό

$$\text{maximize } \sum_{j \in C} \alpha_j$$

$$\sum_{j \in C} \beta_{ij} \leq f_i, \forall i \in F$$

$$\alpha_j - \beta_{ij} \leq d_{i,j}, \forall i \in F, j \in C$$

$$\alpha_j, \beta_{ij} \geq 0$$

Είδη μεταβλητών στο πρωτεύον και στο δυϊκό

- Πρωτεύουσες μεταβλητές y_i εκφράζουν αν ανοίξαμε υπηρεσία στο $i \in F$.
- Πρωτεύουσες μεταβλητές x_{ij} εκφράζουν αν αναθέσαμε υπηρεσία j στο i .

Είδη μεταβλητών στο πρωτεύον και στο δυϊκό

- Πρωτεύουσες μεταβλητές y_i εκφράζουν αν ανοίξαμε υπηρεσία στο $i \in F$.
- Πρωτεύουσες μεταβλητές x_{ij} εκφράζουν αν αναθέσαμε υπηρεσία j στο i .
- Δυϊκές μεταβλητές α_j εκφράζουν 'χρέος' για τον πελάτη $j \in C$.

Είδη μεταβλητών στο πρωτεύον και στο δυϊκό

- Πρωτεύουσες μεταβλητές y_i εκφράζουν αν ανοίξαμε υπηρεσία στο $i \in F$.
- Πρωτεύουσες μεταβλητές x_{ij} εκφράζουν αν αναθέσαμε υπηρεσία j στο i .
- Δυϊκές μεταβλητές α_j εκφράζουν 'χρέος' για τον πελάτη $j \in C$.
- Δυϊκές μεταβλητές β_{ij} εκφράζουν 'ελλάτωση χρεους' (έκπτωση) για την ανάθεση $i \rightarrow j$

Είδη μεταβλητών στο πρωτεύον και στο δυϊκό

- Πρωτεύουσες μεταβλητές y_i εκφράζουν αν ανοίξαμε υπηρεσία στο $i \in F$.
 - Πρωτεύουσες μεταβλητές x_{ij} εκφράζουν αν αναθέσαμε υπηρεσία j στο i .
 - Δυϊκές μεταβλητές α_j εκφράζουν 'χρέος' για τον πελάτη $j \in C$.
 - Δυϊκές μεταβλητές β_{ij} εκφράζουν 'ελλάτωση χρεους' (έκπτωση) για την ανάθεση $i \rightarrow j$.
- 👉 Ο περιορισμός $\alpha_j - \beta_{ij} \leq d_{i,j}$ εκφράζει ότι το συνολικό χρέος για την ανάθεση $j \rightarrow i$ πρέπει να είναι $\leq d_{i,j}$.

Είδη μεταβλητών στο πρωτεύον και στο δυϊκό

- Πρωτεύουσες μεταβλητές y_i εκφράζουν αν ανοίξαμε υπηρεσία στο $i \in F$.
 - Πρωτεύουσες μεταβλητές x_{ij} εκφράζουν αν αναθέσαμε υπηρεσία j στο i .
 - Δυϊκές μεταβλητές α_j εκφράζουν 'χρέος' για τον πελάτη $j \in C$.
 - Δυϊκές μεταβλητές β_{ij} εκφράζουν 'ελλάτωση χρεους' (έκπτωση) για την ανάθεση $i \rightarrow j$.
- 👉 Ο περιορισμός $\alpha_j - \beta_{ij} \leq d_{i,j}$ εκφράζει ότι το συνολικό χρέος για την ανάθεση $j \rightarrow i$ πρέπει να είναι $\leq d_{i,j}$.
- 👉 Ο περιορισμός $\sum_{j \in C} \beta_{ij} \leq f_i$ εκφράζει ότι για κάθε υπηρεσία έχω μια συγκεκριμένη συνολική ελάττωση χρέους που έχω να κατανείμω.