ΚΕΦΑΛΑΙΟ 4

ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΑΚΕΡΑΙΟ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ

4.1 ΕΙΣΑΓΩΓΗ

Σ' αυτό το κεφάλαιο θα μελετήσουμε τις βασικές αρχές της διακριτής βελτιστοποίησης (discrete optimization). Με τον όρο αυτό εννοούμε προβλήματα βελτιστοποίησης όπου η εφικτή περιοχή είναι διακριτό (πεπερασμένο ή αριθμήσιμο) σύνολο. Τα προβλήματα διακριτής βελτιστοποίησης είναι γενικά δυσκολότερα από τα "αντίστοιχα" συνεχή προβλήματα. Αυτός βέβαια ο ισχυρισμ δεν έχει και πολύ νόημα προς το παρόν, καθώς δεν έχουμε ορίσει τα προβλήματα αυτής της κατηγορίας ούτε τα αντίστοιχά τους συνεχ Επίσης υπάρχουν εξαιρέσεις και των δύο τύπων, δηλαδή πολλά διακριτά προβλήματα είναι εύκολα, ενώ πολλά συνεχή δύσκολα.

Εδώ θα εξετάσουμε μια κατηγορία προβλημάτων διακριτής βελτιστοποίησης, τα προβλήματα ακέραιου προγραμματισμού (integer programming). Αυτά προκύπτουν από προβλήματα γραμμικού προγραμματισμού με κάποιες από τις μεταβλητές να περιορίζονται σε ακέραιες τιμές. Θα ορίσουμε διάφορες κατηγορίες προβλημάτων ακέραιου προγραμματισμού, και θα εξετάσουμε τις σχέσεις μεταξύ τους. Επίσης θα δούμε αρκετές περιπτώσεις προβλημάτων αποφάσεων που μοντελοποιούνται με τη βοήθεια ακέραιου προγραμματισμού. Τέλος θα συζητήσουμε μερικές βασικές ιδιότητες των προβλημάτων αυτού του τύπου και μια αρκετά γενική μέθοδο επίλυσης.

Το γενικό πρόβλημα βελτιστοποίησης που θα μελετήσουμε είναι το πρόβλημα του <u>Μεικτού Ακέραιου Προγραμματισμού</u> (mixed integer programming), σε συντομογραφία μ.α.π (MIP). Αυτό ορίζεται ως εξής, σε κανονική μορφή

$$z = \max \ \underline{c}'\underline{x} + \underline{h}'\underline{y}$$

$$A\underline{x} + G\underline{y} = \underline{b} \qquad (4.1)$$

$$\underline{x} \in \mathbb{Z}^n_+, \ \underline{y} \in \mathbb{R}^p_+$$

όπου ο A είναι πίνακας $m \times n$, G πίνακας $m \times p$, $\underline{c} \in \mathbb{R}^n$, $\underline{h} \in \mathbb{R}^p$ και $\underline{b} \in \mathbb{R}^m$.

Η (4.1) εκφράζει ένα πρόβλημα μεγιστοποίησης μιας γραμμικής συνάρτησης κάτω από ένα σύνολο γραμμικών εξισώσεων και περιορισμών μη αρνητικότητας, ακριβώς όπως και σε ένα πρόβλημα γραμμικού προγραμματισμού. Το επιπλέον στοιχείο είναι ότι οι n από τις συνολικά $n+p$ μεταβλητές απόφασεις απαιτείται να παίρνουν ακέραιες τιμές.

Πριν προχωρήσουμε σημειώνουμε ότι στη γενική μορφή του, ένα πρόβλημα μ.α.π. μπορεί να είναι min ή max, να περιέχει περιορισμούς με τη μορφή γραμμικών ανισοτήτων, ενώ κάποιες από τις μεταβλητές $\underline{x}$ ή/και $\underline{y}$ να είναι είτε μη θετικές είτε χωρίς περιορισμό ως προς το πρόσημο.

Με εντελώς ανάλογους μετασχηματισμούς όπως και για ένα π.γ.π. μπορεί κανείς να μετατρέψει ένα γενικό πρόβλημα μ.α.π. σε ένα ισοδύναμο μ.α.π. σε κανονική μορφή. Επομένως η έκφραση σε κανονική μορφή όπως στην (4.1) γίνεται χωρίς βλάβη της γενικότητας.

. Από τη γενική περίπτωση ενός προβλήματος μ.α.π. μπορούμε να δούμε ότι προκύπτουν αρκετές ενδιαφέρουσες κατηγορίες προβλημάτων ως ειδικές περιπτώσεις.

(1) Αν $n=0$ τότε το πρόβλημα

$$z = \max \underline{h}'\underline{y}$$
$$G\underline{y} = \underline{b}$$
$$\underline{y} \in \mathbb{R}_+^p$$

είναι ένα πρόβλημα <u>γραμμικού Προγραμματισμού</u>

(2) Αν $p=0$, το πρόβλημα

$$z = \max \underline{c}'\underline{x}$$
$$A\underline{x} = \underline{b}$$
$$\underline{x} \in \mathbb{Z}_+^n$$

είναι ένα πρόβλημα <u>καθαρού ακέραιου προγραμματισμού</u> όπου όλες οι μεταβλητές είναι περιορισμένες σε ακέραιες τιμές.

Παρατηρούμε ότι ένα γενικό πρόβλημα ακέραιου προγραμματισμού δεν μπορεί γενικά να μετασχηματιστεί σε ισοδύναμο πρόβλημα πάλι ακέραιου προγραμματισμού σε κανονική μορφή. Για παράδειγμα ένας περιορισμός της μορφής $\sqrt{2}x_1 + x_2 \le 4$, όπου $x_1, x_2 \in \mathbb{Z}_+$ μπορεί να γραφεί ισοδύναμα ως $\sqrt{2}x_1 + x_2 + x_3 = 4$, με $x_1, x_2 \in \mathbb{Z}_+$, αλλά $x_3 \in \mathbb{R}_+$. Δεν υπάρχει ισοδύναμη έκφραση ως εξίσωση με όλες τις μεταβλητές ακέραιες.

Συνήθως όμως σε προβλήματα ακέραιου προγραμματισμού γίνεται η επιπλέον υπόθεση ότι οι πίνακες A, G και τα διανύσματα $\underline{c}, \underline{h}, \underline{b}$ αποτελούνται από ρητούς αριθμούς. Αυτή η υπόθεση ρητότητας των δεδομένων διευκολύνει σημαντικά τη θεωρητική ανάλυση, ενώ από πρακτική άποψη δεν είναι σχεδόν καθόλου περιοριστική, δεδομένου ότι στη μεγάλη πλειονότητα των εφαρμογών αυτού του τύπου όλα τα δεδομένα εκτιμούνται με πεπερασμένη ακρίβεια.

Τώρα κάτω από την υπόθεση ρητότητας είναι εύκολο να δει κανείς ότι οποιοδήποτε πρόβλημα καθαρού ακέραιου προγραμματισμού μπορεί να μετασχηματιστεί σε ισοδύναμο πρόβλημα επίσης καθαρού ακέραιου προγραμματισμού σε κανονική μορφή. Επομένως και σε αυτή την κατηγορία προβλημάτων η έκφραση σε κανονική μορφή γίνεται χωρίς βλάβη της γενικότητ(ας)
(Η απόδειξη του ισχυρισμού αφήνεται ως άσκηση.
Η βασική ιδέα είναι ότι ένας περιορισμός της μορφής π.χ.
$\frac{1}{2}x_1+\frac{4}{3}x_2 \le 2$, $x_1, x_2 \in \mathbb{Z}_+$ μπορεί να γραφτεί ισοδύναμ(α)

$3x_1+8x_2 \le 12$, $x_1, x_2 \in \mathbb{Z}_+$, και αυτός με τη σειρά του ως

$3x_1+8x_2+x_3 = 12$, $x_1, x_2, x_3 \in \mathbb{Z}_+$, που είναι σε κανονική μορφή με ακέραιες μεταβλητές).

③ Ένα πρόβλημα της μορφής

$$z = \max \underline{c}'\underline{x}$$
$$A\underline{x} = \underline{b}$$
$$x_j \in \{0,1\},\ j=1,\dots,n$$

ονομάζεται πρόβλημα <u>0-1 προγραμματισμού</u> (zero-one or binary programming). Θέτοντας $\mathbb{B}=\{0,1\}$

το πρόβλημα γράφεται ισοδύναμα

$$z = \max \underline{c}'\underline{x}$$
$$A\underline{x} = \underline{b}$$
$$\underline{x} \in \mathbb{B}^n.$$

Είναι εύκολο να δει κανείς ότι ένα πρόβλημα 0-1 προγραμματισμού είναι ειδική περίπτωση του ακέραιου προγραμματισμού. Πραγματικά ο περιορισμός $x_j \in \{0,1\}$ είναι ισοδύναμος με $x_j \leq 1$ και $x_j \in \mathbb{Z}_+$, από τους οποίους ο πρώτος μπορεί να συμπεριληφθεί στους περιορισμούς $A\underline{x} = \underline{b}$.

Ο λόγος που τα προβλήματα 0-1 προγραμματισμού αξίζουν ειδική μνεία είναι ότι οι δυαδικές μεταβλητές $x_j \in \{0,1\}$ είναι εξαιρετικά χρήσιμες για τη μοντελοποίηση μιας μεγάλης κατηγορίας εφαρμογών ως προβλημάτων ακέραιου προγραμματισμού. Πρόκειται για εφαρμογές όπου πρέπει να ληφθούν αποφάσεις διακριτής μορφής, όχι απαραίτητα όλες ποσοτικές. Λεπτομέρειες και παραδείγματα τέτοιων μοντέλων θα εξετάσουμε στην επόμενη ενότητα.

Σημειώνουμε επίσης ότι μπορεί κανείς να ορίσει προβλήματα της μορφής (4.1) όπου κάποιες από τις μεταβλητές είναι ακέραιες, κάποιες 0-1, και κάποιες πραγματικές. Από την παραπάνω συζήτηση προκύπτει ότι και αυτά τα "γενικότερα" προβλήματα εμπίπτουν στην κατηγορία των προβλημάτων μ.α.π. της μορφής (4.1).

Τέλος μπορεί κανείς να δείξει (βλ. Άσκηση **) ότι αν η εφικτή περιοχή ενός προβλήματος ακέραιου προγραμματισμού είναι φραγμένο σύνολο, τότε το πρόβλημα μπορεί να εκφραστεί ισοδύναμα ως πρόβλημα 0-1 προγραμματισμού. Επομένως τα προβλήματα 0-1 προγραμματισμού είναι στην πραγματικότητα αρκετά γενικότερα απ' ότι φαίνεται με την πρώτη ματιά.

4.2. ΕΦΑΡΜΟΓΕΣ ΑΚΕΡΑΙΟΥ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

Στην ενότητα αυτή θα παρουσιάσουμε μερικές κατηγορίες εφαρμογών που μοντελοποιούνται ως προβλήματα μεικτού ακέραιου προγραμματισμού, και ιδιαίτερα χρησιμοποιώντας δυαδικές μεταβλητές.

Πριν εξετάσουμε συγκεκριμένα προβλήματα, αναφέρουμε ότι πολλά προβλήματα γραμμικού προγραμματισμού (όπως αυτά που είδαμε στο Κεφάλαιο 1) μετατρέπονται αυτόματα σε προβλήματα μ.α.π. αν από τη φύση της εφαρμογής κάποιες από τις μεταβλητές περιορίζονται σε ακέραιες τιμές. Για παράδειγμα προβλήματα προγραμματισμού παραγωγής είναι προβλήματα μ.α.π. αν κάποια προϊόντα είτε είναι από τη φύση τους διακριτά (π.χ. αυτοκίνητα) είτε απαιτείται η παραγωγή σε διακριτές ποσότητες (π.χ. ακέραιος αριθμός βαρελιών κρασιού).

Αυτές οι περιπτώσεις δεν παρουσιάζουν ιδιαίτερο ενδιαφέρον όσον αφορά τη μοντελοποίηση, καθώς δεν απαιτούν νέες ιδέες σε σχέση με τα αντίστοιχα π.γ.π. Εδώ θα ασχοληθούμε με κατηγορίες μοντέλων που επεκτείνουν ουσιαστικά το σύνολο των εφαρμογών.

Η βασική ιδέα στα προβλήματα που θα εξετάσουμε είναι η χρήση δυαδικών μεταβλητών απόφασης για να εκφράσουμε διχοτομικές αποφάσεις του τύπου ναι ή όχι.
Έτσι μια δυαδική μεταβλητή $x_j \in \{0,1\}$ θα χρησιμοποιείται για να εκφράσει την επιλογή μιας συγκεκριμένης απόφασης-j:

$$x_j = \begin{cases} 1, & \text{αν επιλέξουμε την απόφαση} - j \\ 0, & \text{διαφορετικά} \end{cases},$$

ή ισοδύναμα ως δείκτρια μεταβλητή της απόφασης-j

$$x_j = 1(\text{απόφαση} - j) \tag{4.2}$$

Με βάση αυτό τον ορισμό μπορούμε να δούμε κάποιους γενικούς κανόνες μοντελοποίησης με δυαδικές μεταβλητές, που είναι εφαρμόσιμοι σε πολλά διαφορετικά προβλήματα.

Έστω ένα σύνολο δυνατών αποφάσεων $1,\dots,n$, κάθε μια από τις οποίες μπορεί να ενεργοποιηθεί ή όχι (δυαδικής μορφής). Ορίζοντας μεταβλητές x_j, $j=1,\dots,n$ όπως στην (4.2) προκύπτουν εύκολα τα παρακάτω:

(i) Η έκφραση $\sum_{j=1}^{n} x_j$ δείχνει τον αριθμό των αποφάσεων που ενεργοποιούνται. Αυτό επιτρέπει τη μοντελοποίηση περιορισμών της μορφής "το πολύ k-αποφάσεις μπορούν να ενεργοποιηθούν":

$$\sum_{j=1}^{n} x_j = k ,$$

κλ.π.

(ii) Ένας περιορισμός της μορφής

$$x_i \leq x_j$$

εκφράζει το γεγονός ότι η απόφαση-i συνεπάγεται την j, δηλαδή μπορεί να ενεργοποιηθεί μόνο αν ενεργοποιηθεί και η j ταυτόχρονα.

(Φυσικά αν η j είναι ικανή και αναγκαία για την i, αυτό εκφράζεται με τον περιορισμό $x_i = x_j$, οπότε μια από τις δύο ισοδύναμες αποφάσεις/μεταβλητές θα μπορούσε να απαλειφθεί.)

(iii) Έστω ότι χρειάζεται να ορίσουμε μια μεταβλητή για το γεγονός ότι οι αποφάσεις i και j επιλέγονται μαζί:

$$x_{ij} = \begin{cases} 1, & \text{αν } i \text{ και } j \text{ λαμβάνονται ταυτόχρονα} \\ 0, & \text{διαφορετικά.} \end{cases}$$

Χρησιμοποιώντας ιδέες από την άλγεβρα Boole θα μπορούσαμε να ορίσουμε την $x_{ij} = x_i x_j$. Παρ' όλο που αυτό είναι μαθηματικά σωστό, δεν είναι το κατάλληλο μοντέλο, καθώς η έκφραση $x_i x_j$ <u>δεν είναι γραμμική</u>. Μια ισοδύναμη γραμμική μοντελοποίηση είναι με τρείς περιορισμούς:

$$x_{ij} \leq x_i$$
$$x_{ij} \leq x_j$$
$$x_{ij} \geq x_i + x_j - 1$$
$$x_i, x_j, x_{ij} \in \mathbb{B}$$

Μπορεί κανείς να επαληθεύσει ότι από τους παραπάνω περιορισμούς προκύπτει $x_{ij} = 1$ αν και μόνο αν $x_i = x_j = 1$ (φυσικά κάτω από την υπόθεση $x_{ij}, x_i, x_j \in \mathbb{B}$)

Μπορούμε τώρα να δούμε κάποιες ενδιαφέρουσες κατηγορίες προβλημάτων βελτιστοποίησης στις οποίες βρίσκουν εφαρμογή οι δυαδικές μεταβλητές.

4.2.1. Το πρόβλημα τοποθέτησης σταθμών παραγωγής

Στη γενική περίπτωση το πρόβλημα τοποθέτησης σταθμών παραγωγής (facility location problem) περιγράφεται ως εξής: Δίνονται n δυνατές τοποθεσίες στις οποίες μπορούν να αναπτυχθούν σταθμοί παραγωγής ή εξυπηρέτησης μιας εταιρείας. Επίσης υπάρχουν m πελάτες που πρέπει να εξυπηρετηθούν. Υπάρχει ένα κόστος c_j αν ανοίξει σταθμός στην τοποθεσία j, $j=1,\ldots,n$. Επίσης υπάρχει κόστος h_{ij} αν ο πελάτης i σταλεί στο σταθμό της τοποθεσίας j για εξυπηρέτηση. Κάθε πελάτης πρέπει να εξυπηρετηθεί από ένα μοναδικό σταθμό. Το πρόβλημα είναι να βρεθεί σε ποιές τοποθεσίες θα ανοίξουν σταθμοί και πώς θα γίνει η κατανομή των πελατών στους σταθμούς ώστε το συνολικό κόστος να είναι ελάχιστο.

Εδώ έχουμε δύο τύπων αποφάσεις και χρησιμοποιούμε δύο κατηγορίες δυαδικών μεταβλητών απόφασης:

$$x_j = 1 \text{ (ανοίγει σταθμός στην τοποθεσία } j\text{)}, \quad j=1,\ldots,n$$

$$y_{ij} = 1 \text{ (ο πελάτης } i \text{ εξυπηρετείται από την τοποθεσία } j\text{)}, \quad i=1,\ldots,m,\ j=1,\ldots,n$$

Με βάση αυτή την επιλογή μεταβλητών, η αντικειμενική συνάρτηση για το συνολικό κόστος είναι

$$\text{Συνολικό κόστος} = \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \sum_{j=1}^{n} h_{ij} y_{ij}$$

Για τους περιορισμούς έχουμε: Πρώτον ο πελάτης i μπορεί να εξυπηρετηθεί από την τοποθεσία j μόνο αν υπάρχει σταθμός σ' αυτή την τοποθεσία. Επομένως σύμφωνα με όσα είδαμε παραπάνω έχουμε περιορισμούς της μορφής

$$y_{ij} \le x_j, \quad i=1,\ldots,m, \quad j=1,\ldots,n$$

Επίσης κάθε πελάτης πρέπει να εξυπηρετηθεί από ακριβώς ένα σταθμό, δηλαδή

$$\sum_{j=1}^{n} y_{ij} = 1, \quad i=1,\ldots,m$$

Περιληπτικά το μοντέλο ακέραιου προγραμματισμού για το παραπάνω πρόβλημα είναι

$$\min \sum_{j=1}^{n} c_j x_j + \sum_{i=1}^{m} \sum_{j=1}^{n} h_{ij} y_{ij}$$

$$y_{ij} \le x_j, \quad i=1,\ldots,m, \; j=1,\ldots,n$$

$$\sum_{j=1}^{n} y_{ij} = 1, \quad i=1,\ldots,m$$

$$x_j, y_{ij} \in \mathbb{B} \quad \forall i, j.$$

4.2.2. Το πρόβλημα τοποθέτησης - Παραγωγής.

Το πρόβλημα αυτό αποτελεί παραλλαγή του προηγούμενου. Έστω ότι η εταιρεία παράγει ένα προϊόν, από το οποίο χρειάζεται συνολική ποσότητα d σε ετήσια βάση. Για την παραγωγή μπορεί να ανοίξει εργοστάσια σε οποιεσδήποτε από n δεδομένες τοποθεσίες, $j=1,\dots,n$. Αν ανοίξει σταθμός στην τοποθεσία j τότε υπάρχει ένα ετήσιο κόστος συντήρησης του σταθμού ίσο με K_j. Επίσης η δυναμικότητα του σταθμού (δηλ. η μέγιστη ετήσια ποσότητα παραγωγής) είναι ίση με M_j, ενώ το μοναδιαίο κόστος παραγωγής για το συγκεκριμένο προϊόν ίσο με K_j. Το πρόβλημα είναι να βρεθεί πού θα ανοίξουν σταθμοί όπως επίσης και οι αντίστοιχες ποσότητες παραγωγής, ώστε να ικανοποιηθεί η συνολική ετήσια ζήτηση με το ελάχιστο συνολικό κόστος.

Ορίζουμε τις παρακάτω μεταβλητές απόφασης:

$x_j = 1$ (ανοίγει σταθμός στην τοποθεσία j), $j=1,\dots,n$ $(\in \mathbb{B})$

$y_j =$ ποσότητα παραγωγής στην τοποθεσία j $(\in \mathbb{R}_+)$

Το συνολικό κόστος είναι ίσο με

$$\sum_{j=1}^{n} K_j x_j + \sum_{j=1}^{n} h_j y_j$$

Για τους περιορισμούς έχουμε: Πρώτον η συνολική ποσότητα παραγωγής πρέπει να είναι ίση με d:

$$\sum_{j=1}^{n} y_j = d$$

Επίσης αν στην τοποθεσία j δεν ανοίξει εργοστάσιο (σταθμός) τότε δεν μπορεί να γίνει παραγωγή, ενώ αν ανοίξει εργοστάσιο, τότε η παραγωγή δεν μπορεί να υπερβαίνει τη δυναμικότητα M_j. Αλγεβρικά αυτό σημαίνει ότι αν $x_j = 0$ τότε $y_{ij} = 0$, ενώ αν $x_j = 1$ τότε $y_j \le M_j$.

Το παραπάνω εκφράζεται με ένα περιορισμό της μορφής

$$y_j \le M_j x_j$$

Είναι εύκολο να δει κανείς ότι ανάλογα με την τιμή του $x_j = 0$ ή 1 παίρνουμε το σωστό περιορισμό για τό y_{ij}.

Επίσης παρατηρούμε ότι ο παραπάνω περιορισμός είναι γραμμικός (η M_j είναι δεδομένη σταθερά)

Συνοπτικά καταλήγουμε στο παρακάτω πρόβλημα μ.α.π.

$$\min \sum_{j=1}^{n} k_j x_j + \sum_{j=1}^{n} h_j y_j$$

$$\sum_{j=1}^{n} y_j = d$$

$$y_j \le M_j x_j \quad , \quad j=1,\dots,n$$

$$x_j \in \mathbb{B}, \quad j=1,\dots,n$$

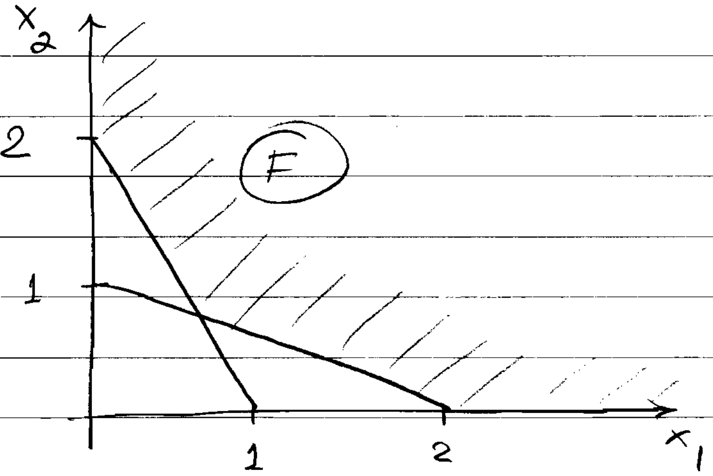
$$y_j \ge 0, \; y_j \in \mathbb{R}, \quad j=1,\dots,n$$

4.2.3 Προβλήματα με Διαζευκτικούς Περιορισμούς

Σε όλα τα προβλήματα μαθηματικού προγραμματισμού (γραμμικού, μη γραμμικού και ακέραιου) που έχουμε μελετήσει υποθέτουμε πάντα ότι όλοι οι περιορισμοί πρέπει να ικανοποιούνται, με άλλα λόγια μια λύση είναι εφικτή αν ικανοποιεί όλους τους περιορισμούς.
Σε ορισμένες περιπτώσεις όμως υπάρχουν περιορισμοί διαζευκτικού τύπου, απαιτείται δηλαδή να ικανοποιείται τουλάχιστον ένας και όχι αναγκαστικά όλοι.
Για παράδειγμα ας θεωρήσουμε ~~ένα~~ συνηθισμένο πρόβλημα γραμμικού προγραμματισμού με εφικτή περιοχή

$$F = \{ (x_1, x_2) \mid x_1 + 2x_2 \geq 2,\ 2x_1 + x_2 \geq 2,\ x_1, x_2 \geq 0 \}$$

που παριστάνεται γραφικά στο σχήμα 4.1.

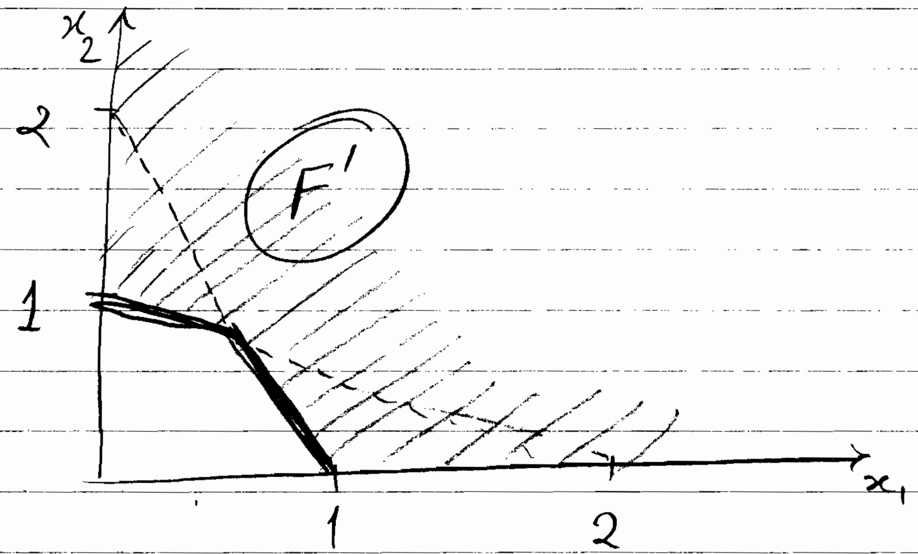


Σχήμα 4.1

Ας υποθέσουμε τώρα ότι μια λύση είναι εφικτή αν ικανοποιεί τουλάχιστον ένα από τους περιορισμούς, δηλαδή αν είτε $2x_1 + x_2 \geq 2$ ή $x_1 + 2x_2 \geq 2$ ή και τα δύο.

Τώρα η εφικτή περιοχή F' είναι υπερσύνολο της F

και παρουσιάζεται στο Σχήμα 4.2



Σχήμα 4.2

$$F' = \{(x_1,x_2) \mid 2x_1+x_2 \geq 2 \text{ ή } x_1+2x_2 \geq 2,\quad x_1,x_2 \geq 0\}$$

Στο συγκεκριμένο παράδειγμα παρατηρούμε ότι η εφικτή περιοχή F δεν είναι κυρτό σύνολο, επομένως δεν είναι δυνατό να εκφραστεί ως πρόβλημα γραμμικού προγραμματισμού.

Χρησιμοποιώντας δυαδικές μεταβλητές μπορούμε να εκφράσουμε αυτού του είδους τα προβλήματα με τη βοήθεια 0-1 προγραμματισμού. Για παράδειγμα για το σύνολο F' εισάγουμε μια δυαδική μεταβλητή y και θεωρούμε τους εξής περιορισμούς

$$\begin{aligned} 2x_1+x_2 &\geq 2y \\ x_1+2x_2 &\geq 2y \qquad (4.3) \\ x_1,x_2 \geq 0,&\quad y\in\{0,1\} \end{aligned}$$

Που απαιτούμε να ικανοποιούνται όλοι, όπως στα συνήθη προβλήμα

Για να δούμε ότι αυτή η μορφοποίηση είναι ισοδύναμη με το αρχικό πρόβλημα, θεωρούμε το σύνολο

$$\tilde{F} = \{(x_1, x_2, y) \mid \text{ικανοποιούνται οι } (4.3)\}$$

Τότε $\tilde{F} = \{(x_1,x_2,0) \mid (4.3)\} \cup \{(x_1,x_2,1) \mid (4.3)\}$

$$\tilde{F} = \{(x_1,x_2,0) \mid 2x_1+x_2 \geq 0,\ x_1+2x_2 \geq 2,\ x_1,x_2 \geq 0\} \cup$$

$$\{(x_1,x_2,1) \mid 2x_1+x_2 \geq 2,\ x_1+2x_2 \geq 0,\ x_1,x_2 \geq 0\}$$

Επειδή $2x_1+x_2 \geq 0 \ \forall\, x_1,x_2 \geq 0$ και $x_1+2x_2 \geq 0 \ \forall\, x_1,x_2 \geq 0 \Rightarrow$

$$\tilde{F} = \{(x_1,x_2,0) \mid x_1+2x_2 \geq 2,\ x_1,x_2 \geq 0\} \cup \{(x_1,x_2,1) \mid 2x_1+x_2 \geq 2,\ x_1,x_2 \geq 0\}$$

δηλαδή υπάρχει ένα προς ένα αντιστοιχία ανάμεσα στα στοιχεία του συνόλου F' και του $\tilde{F}$.

Διαισθητικά η παρουσία της y επιτρέπει το εξής: αν $y=1$ τότε επιβάλλεται ο περιορισμός $x_1+2x_2 \geq 2$, ενώ ο πρώτος γίνεται πάντα αληθής δηλαδή δεν εξετάζεται αν ικανοποιείται ή όχι. Αντίστοιχα για $y=0$.

Παρ' όλο που το παραπάνω τέχνασμα δουλεύει για το συγκεκριμένο πρόβλημα, δεν γενικεύεται εύκολα.
Για παράδειγμα, τι γίνεται αν έχουμε 3 περιορισμούς με διάζευξη; Επίσης τι γίνεται αν ο πρώτος π.χ. περιορισμός γίνει $2x_1+x_2 \leq 2$; Τότε δεν μπορούμε

να χρησιμοποιήσουμε το μετασχηματισμένο περιορισμό

$$2x_1 + x_2 \leq 2y \quad ,$$

γιατί για $y=1$ αυτός αντιστοιχεί στον αρχικό $2x_1+x_2 \leq 2$, ενώ για $y=0$ γίνεται $2x_1+x_2 \leq 0$ πού είναι αδύνατος.

Επομένως χρειαζόμαστε μια γενίκευση της προηγούμενης μεθόδου που να λειτουργεί για όλες τις περιπτώσεις.

Έστω ότι έχουμε ένα σύνολο περιορισμών

$$\underline{a}_j' \underline{x} \leq b_j \quad , \quad j=1,\ldots,n \qquad (4.4)$$

από τους οποίους απαιτείται να ικανοποιούνται τουλάχιστον k, για κάποιο $k \leq n$.

Η γενική ιδέα για ενα 0-1 μοντέλο του παραπάνω προβλήματος είναι η εξής:

Εισάγουμε n δυαδικές μεταβλητές y_j, $j=1,\ldots,n$ τέτοιες ώστε

$$y_j = 1 \text{(ο περιορισμός } j \text{ επιβάλλεται να ικανοποιείται)}$$

Τότε οι περιορισμοί πρέπει να εκφραστούν με τη βοήθεια των y_j έτσι ώστε αν $y_j=1$ ο j- περιορισμός ισχύει όπως έχει δοθεί ενώ αν $y_j=0$ αντιστοιχει σε τετριμμένο περιορισμό, δηλαδή σε ταυτότητα που ισχύει πάντα. Αυτό γίνεται αν γράφουμε

$$\underline{a}_j' \underline{x} \leq b_j y + M(1-y),$$

όπου $M \to \infty$ είναι μια αυθαίρετα μεγάλη σταθερά.

Παρατηρούμε ότι για $y_j = 1 \Rightarrow \underline{a}_j'\underline{x} \leq b_j$ ενώ για $y_j = 0 \Rightarrow \underline{a}_j'\underline{x} \leq M$ πού είναι τετριμμένος περιορισμός.

(δηλαδή για $y_j = 0$ ο περιορισμός δεν είναι πια περιορισμός επειδή το $\underline{a}_j'\underline{x}$ επιτρέπεται να παίρνει οποιαδήποτε τιμή).

Για να συμπληρωθεί το μοντέλο πρέπει τουλάχιστον k από τους n περιορισμούς να ικανοποιηθούν. Επομένως χρειαζόμαστε

$$y_1 + y_2 + \cdots + y_n \geq k$$

Συνοπτικά το παρακάτω μοντέλο είναι ισοδύναμο με το αρχικό (4.4):

$$\begin{aligned} &\underline{a}_j'\underline{x} \leq b_j + M(1-y_j) \quad j=1,\ldots,n \\ &\sum_{j=1}^{n} y_j \geq k, \end{aligned} \qquad (4.5)$$

Επισημαίνουμε ότι το μοντέλο (4.5) αντιστοιχεί στην εφικτή περιοχή ενός προβλήματος μεικτού ακέραιου προγραμματισμού, καθώς απαιτείται να ισχύουν όλοι οι περιορισμοί συζευκτικά.
Αυτό που κερδίζουμε με την εισαγωγή των δυαδικών μεταβλητών είναι να εκφράσουμε το διαζευκτικό σύνολο (4.4) ως συζευκτικό σύνολο (4.5).

4.2.4. Μεταβλητές με πεπερασμένο σύνολο τιμών

Έστω ότι σε ένα πρόβλημα βελτιστοποίησης μια μεταβλητή απόφασης x απαιτείται να παίρνει τιμές σε ένα πεπερασμένο σύνολο:

$$x \in \{a_1, a_2, \dots, a_k\} \tag{4.6}$$

Περιορισμοί αυτού του τύπου είναι γενικότεροι από τους περιορισμούς ακέραιων τιμών. Για παράδειγμα ο περιορισμός

$$x \in \{2, 3, 4, 5\}$$

μπορεί να εκφραστεί ως

$$x \le 5$$
$$x \ge 2$$
$$x \in \mathbb{Z},$$

όμως ο περιορισμός $x \in \{1/2, 2, 5\}$ δεν μπορεί να εκφραστεί με αντίστοιχο τρόπο.

Για τη γενική περίπτωση (4.6) μπορούμε να εισάγουμε δυαδικές μεταβλητές

$$y_j = 1(x = a_j), \quad j = 1, 2, \dots, k$$

οπότε η (4.6) είναι ισοδύναμη με την

$$x = a_1 y_1 + \dots + a_k y_k$$
$$y_1 + \dots + y_k = 1,$$
$$y_1, \dots, y_k \in \mathbb{B}.$$

4.2.5. Γενική Τμηματικά Γραμμική Αντικειμενική Συνάρτηση

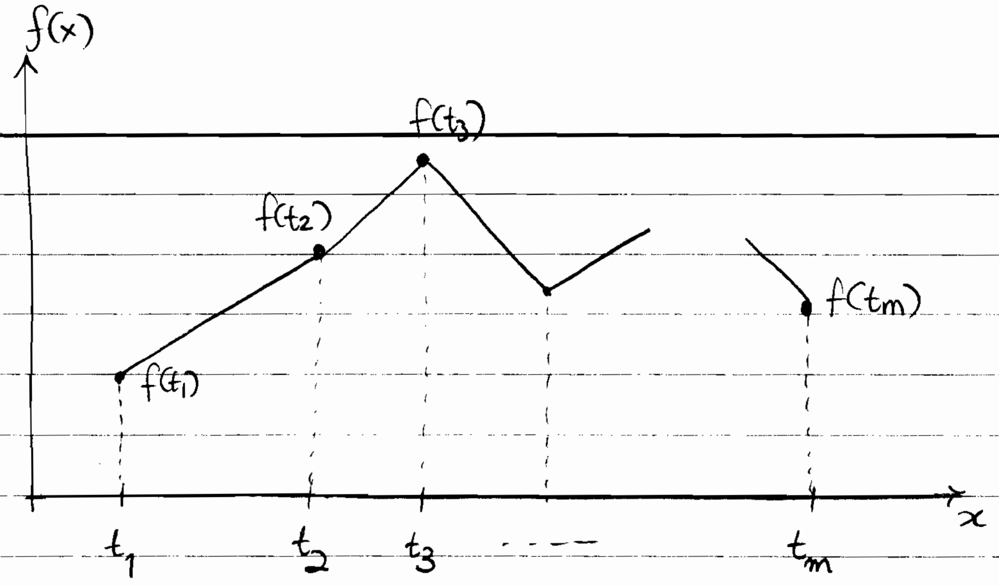
Στο Κεφάλαιο 1 μελετήσαμε διάφορα προβλήματα με τμηματικά γραμμική αντικειμενική συνάρτηση (ενότητα 1.5) τα οποία μπορούν να εκφραστούν ισοδύναμα ως προβλήματα γραμμικού προγραμματισμού. Τέτοια είναι τα προβλήματα μεγιστοποίησης μιας τμηματικά γραμμικής κοίλης ή ελαχιστοποίησης μιας τμηματικά γραμμικής κυρτής συνάρτησης (maximin και minimax αντίστοιχα).

Στη γενική περίπτωση τμηματικά γραμμικής αντικειμενικής συνάρτησης δεν είναι δυνατή η έκφραση ως πρόβλημα γραμμικού προγραμματισμού. Είναι όμως δυνατή η αναγωγή σε πρόβλημα 0-1 προγραμματισμού με τη βοήθεια κατάλληλων δυαδικών μεταβλητών.

Έστω ότι η αντικειμενική συνάρτηση ορίζεται σε ένα διάστημα $[t_1, t_m]$ και είναι τμηματικά γραμμική.

Επομένως υπάρχουν ενδιάμεσα σημεία $t_1 \le t_2 \le \ldots \le t_m$ τέτοια ώστε η f είναι γραμμική σε κάθε διάστημα $[t_i, t_{i+1}]$, $i = 1, \ldots, m-1$, και επίσης συνεχής στο $[t_1, t_m]$.

Με βάση τα παραπάνω η f προσδιορίζεται μονοσήμαντα από τα σημεία $(t_1, \ldots, t_m)$ και τις τιμές που παίρνει σε αυτά: $(f(t_1), \ldots, f(t_m))$, όπως φαίνεται στο Σχήμα 4.3



Σχήμα 4.3

Δεδομένων των $t_1, \ldots, t_m$, $f(t_1), \ldots, f(t_m)$ η $f(x)$ ορίζεται

$$f(x) = \begin{cases} f(t_1) + [f(t_2) - f(t_1)] \cdot \dfrac{x - t_1}{t_2 - t_1}, & x \in [t_1, t_2] \\ f(t_2) + [f(t_3) - f(t_2)] \cdot \dfrac{x - t_2}{t_3 - t_2}, & x \in [t_2, t_3] \\ \vdots & \\ f(t_{m-1}) + [f(t_m) - f(t_{m-1})] \dfrac{x - t_{m-1}}{t_m - t_{m-1}}, & x \in [t_{m-1}, t_m] \end{cases}$$

Έστω το πρόβλημα βελτιστοποίησης

$$z = \min \{ f(x) : x \in [t_1, t_m] \} \tag{4.7}$$

Για την έκφραση του (4.7) ως προβλήματος μ.α.π. σκεφτόμαστε ως εξής:

Καταρχήν αν $x \in [t_i, t_{i+1}]$ για κάποιο $i \in \{1, \ldots, m-1\}$, τότε το x μπορεί να εκφραστεί μονοσήμαντα ως κυρτός συνδυασμός των t_i, t_{i+1} δηλ. υπάρχει μοναδικό ζεύγος $(\lambda_i, \lambda_{i+1})$ τ.

$$x = \lambda_i t_i + \lambda_i t_{i+1}$$

$$\lambda_i + \lambda_{i+1} = 1$$

$$\lambda_i, \lambda_{i+1} \geq 0$$

Σε αυτή την περίπτωση είναι εύκολο να δει κανείς ότι και η $f(x)$ έχει αντίστοιχη έκφραση:

$$f(x) = \lambda_i f(t_i) + \lambda_{i+1} f(t_{i+1})$$

Ένας ισοδύναμος τρόπος να εκφράσουμε την παραπάνω ιδιότητα είναι να γράψουμε

$$\begin{aligned} x &= \lambda_1 t_1 + \cdots + \lambda_m t_m \\ f(x) &= \lambda_1 f(t_1) + \cdots + \lambda_m f(t_m) \\ \lambda_1 &+ \cdots + \lambda_m = 1 \\ \lambda_1, &\ldots, \lambda_m \geq 0 \end{aligned} \tag{4.8}$$

με τον επιπλέον περιορισμό ότι αν $x \in [t_i, t_{i+1})$ τότε μόνο τα λ_i, λ_{i+1} επιτρέπεται να είναι θετικά ενώ όλα τα υπόλοιπα $\lambda_j = 0$. Για να εκφράσουμε την τελευταία ιδιότητα εισάγουμε $m-1$ δυαδικές μεταβλητές:

$$y_i = 1(x \in [t_i, t_{i+1})) \quad , i = 1, \ldots, m-2$$

$$y_{m-1} = 1(x \in [t_{m-1}, t_m])$$

Αυτές συνδέονται με την (4.8), καθώς η μεταβλητή λ_i επιτρέπεται να είναι θετική μόνο αν $y_{i-1} = 1$ ή $y_i = 1$.

Επομένως μια ισοδύναμη έκφραση του προβλήματος (4.8) είναι

$$\min \sum_{i=1}^{m} \lambda_i f(t_i) \quad (4.9)$$

$$\text{υ.π.} \quad \sum_{i=1}^{m} \lambda_i = 1$$

$$\lambda_1 \le y_1$$

$$\lambda_i \le y_{i-1} + y_i \quad , \quad i = 2, \ldots, m-1$$

$$\lambda_m \le y_{m-1}$$

$$\sum_{i=1}^{m-1} y_i = 1$$

$$\lambda_i \ge 0 \quad , \quad i = 1, \ldots, m$$

$$y_i \in \mathbb{B} , \quad i = 1, \ldots, m-1$$

Στην (4.9) ο περιορισμός $\sum_{i=1}^{m-1} y_i = 1$ εκφράζει το γεγονός ότι το x μπορεί να βρίσκεται σε ακριβώς ένα από τα διαστήματα $[t_1, t_2), [t_2, t_3), \ldots, [t_{m-1}, t_m]$.

Στην παραπάνω ανάλυση δεν αλλάζει τίποτα αν έχουμε μεγιστοποίηση της $f(x)$.

4.2.6. Το Πρόβλημα Ανάθεσης

Το πρόβλημα ανάθεσης (assignment problem) ορίζεται γενικά ως εξής: Υπάρχουν n αντικείμενα και n θέσεις. Πρέπει κάθε αντικείμενο να τοποθετηθεί σε μια θέση και κάθε θέση να δεχθεί ακριβώς ένα αντικείμενο.
Αν το αντικείμενο i τοποθετηθεί στη θέση j, υπάρχει κόστος c_{ij}. Ζητείται να βρεθεί η ανάθεση των αντικειμένων που ελαχιστοποιεί το συνολικό κόστος.
Ορίζουμε $n \times n$ δυαδικές μεταβλητές:

$$x_{ij} = 1 \text{(αντικείμενο } i \text{ ανατίθεται στη θέση } j\text{)}, \quad i,j=1,\dots,n$$

Τότε είναι εύκολο να δει κανείς ότι το πρόβλημα γράφεται

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad , \quad i=1,\dots,n$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad , \quad j=1,\dots,n$$

$$x_{ij} \in \mathbb{B} \quad , \quad i,j=1,\dots,n$$

Παρατηρούμε ότι αν αντικαταστήσουμε τους περιορισμούς $x_{ij} \in \mathbb{B}$ με $x_{ij} \geq 0$ τότε προκύπτει ένα πρόβλημα μεταφοράς

4.3 Η ΜΕΘΟΔΟΣ ΚΛΑΔΟΥ-ΦΡΑΓΜΑΤΟΣ

Σε αυτή την ενότητα θα παρουσιάσουμε την επικρατέστερη, από πρακτική και υπολογιστική άποψη, μέθοδο επίλυσης προβλημάτων μ.α.π. Πρόκειται για τη μέθοδο κλάδου-φράγματος (branch and bound), που είναι μέθοδος γενικού σκοπού, δηλαδή μπορεί να εφαρμοστεί για την επίλυση του γενικού προβλήματος μ.α.π.

Πριν προχωρήσουμε στην ανάπτυξη της μεθόδου κλάδου-φράγματος, χρειάζεται να κάνουμε μια εισαγωγική συζήτηση και να δούμε μερικά αρχικά αποτελέσματα που θα χρησιμοποιηθούν στη συνέχεια.

Κατ' αρχήν πρέπει να τονίσουμε ότι τα προβλήματα ακέραιου προγραμματισμού είναι συνήθως πολύ δυσκολότερα από τα αντίστοιχα προβλήματα γραμμικού προγραμματισμού, τόσο από θεωρητική όσο και από υπολογιστική άποψη. Ενώ για το γραμμικό προγραμματισμό η μέθοδος Simplex (αλλά και άλλες νεότερες μέθοδοι όπως του Ελλειψοειδούς και του εσωτερικού σημείου) μπορεί να λύσει προβλήματα με χιλιάδες μεταβλητές και περιορισμούς σε λογικό χρονικό διάστημα, στον ακέραιο προγραμματισμό η κατάσταση είναι πολύ διαφορετική. Εδώ παίζει κρίσιμο ρόλο η μορφή του προβλήματος. Για ορισμένες κατηγορίες προβλημάτων (π.χ. το πρόβλημα ανάθεσης) υπάρχουν πολύ αποτελεσματικές μέθοδοι εύρεσης της βέλτιστης λύσης. Όμως υπάρχουν άλλες κατηγορίες προβλημάτων για τις οποίες η καλύτερη μέθοδος λύσης αυτή τη στιγμή έχει εκθετική πολυπλοκότητα, δηλαδή ο χρόνος επίλυσης αυξάνει εκθετικά με το μέγεθος του προβλήματος, με αποτέλεσμα προβλήματα ακόμη και μεσαίου μεγέθους (της τάξης των 100 μεταβλητών) να μη μπορούν

να λυθούν ούτε σε διάστημα πολλών ετών ακόμα και με τους ταχύτερους υπολογιστές.
Παράλληλα με την υπολογιστική δυσκολία, τα προβλήματα ακέραιου προγραμματισμού (γενικότερα τα προβλήματα διακριτής βελτιστοποίησης) αποτελούν δύσκολα θεωρητικά προβλήματα, που έχουν μελετηθεί εκτεταμένα και βαθιά τα τελευταία 50 χρόνια. Γενικά η περιοχή της διακριτής βελτιστοποίησης αποτελεί μια από τις πιο πλούσιες, δύσκολες αλλά και ερευνητικά ενεργές περιοχές της επιχειρησιακής έρευνας και των μαθηματικών γενικότερα. Ανάμεσα στα άλλα χρησιμοποιεί εργαλεία από τη θεωρία βελτιστοποίησης, τη συνδυαστική ανάλυση, τη θεωρία αριθμών και τη θεωρία υπολογιστικής πολυπλοκότητας.

Στο κεφάλαιο αυτό δεν θα υπεισέλθουμε στη θεωρία του ακέραιου προγραμματισμού. Θα αρκεστούμε στην παρουσίαση της μεθόδου κλάδου φράγματος η οποία έχει το πλεονέκτημα ότι είναι εύκολα κατανοητή, σχετίζεται άμεσα με το γραμμικό προγραμματισμό και μπορεί να προγραμματιστεί χωρίς μεγάλη δυσκολία.

4.3.1. Χαλαρώσεις και Φράγματα

Θα θεωρήσουμε ένα πρόβλημα ακέραιου προγραμματισμού

$$\begin{aligned} z = \max\ & \underline{c}'\underline{x} \\ & A\underline{x} = \underline{b} \\ & \underline{x} \geq 0 \\ & \underline{x} \in \mathbb{Z}^n \end{aligned} \qquad (4.10)$$

Ο περιορισμός σε προβλήματα ακέραιου προγραμματισμού γίνεται κυρίως για ευκολία στην παρουσίαση. Όπως θα φανεί παρακάτω, όλη η συζήτηση μπορεί να γενικευθεί άμεσα στην περίπτωση του μεικτού ακέραιου προγραμματισμού.

Ονομάζουμε γενικά χαλάρωση (relaxation) ενός προβλήματος βελτιστοποίησης ένα πρόβλημα που προκύπτει από το αρχικό με την απαλοιφή ή τη χαλάρωση κάποιων περιορισμών. Η εφικτή περιοχή ενός προβλήματος χαλάρωσης είναι γενικά υπερσύνολο της εφικτής περιοχής του αρχικού προβλήματος, και η βέλτιστη τιμή εξ ίσου καλή ή καλύτερη από αυτή του αρχικού προβλήματος.

Στην περίπτωση του ακέραιου προγραμματισμού ιδιαίτερα χρήσιμη είναι μια συγκεκριμένη χαλάρωση, η λεγόμενη χαλάρωση γραμμικού προγραμματισμού (γ.π.) (LP relaxation). Αυτή προκύπτει από το αρχικό π.α.π. αν απαλείψουμε τους περιορισμούς ακέραιων τιμών. Επομένως για το π.α.π. (4.10) η χαλάρωση γ.π. ορίζεται ως

$$z_{LP} = \max \underline{c}'\underline{x} \qquad (4.11)$$
$$A\underline{x} = \underline{b}$$
$$\underline{x} \geq 0$$

Φυσικά η (4.11) αντιστοιχεί σε ένα π.γ.π. Επίσης ισχύει

$$z_{LP} \geq z \qquad (4.12)$$

με ισότητα αν και μόνο αν η (4.11) έχει τουλάχιστον μια βέλτιστη λύση $x^*_{LP} \in Z^n$, δηλαδή αν η χαλάρωση γ.π. έχει ακέραιη βέλτιστη λύση. (αποδείξτε τον παραπάνω ισχυρισμό).

Η (4.12) είναι χρήσιμη γιατί παρέχει ένα άνω φράγμα για τη βέλτιστη τιμή z που γενικά μπορεί να είναι πολύ δύσκολο να υπολογιστεί ακριβώς. Ας υποθέσουμε τώρα ότι έχουμε βρεί μια εφικτή λύση $\underline{x}^0$ του Προβλήματος Π.Α.Π. (4.10), δηλαδή ακέραιη. Έστω $z^0 = \underline{c}' \cdot \underline{x}^0$ η τιμή της αντικειμενικής συνάρτησης για αυτή τη λύση. Επειδή η $\underline{x}^0$ δεν είναι γενικά βέλτιστη, ισχύει $z^0 \leq z$. Σε συνδυασμό με την ανισότητα (4.12), έχουμε καταφέρει να βρούμε ένα διάστημα μέσα στο οποίο κυμαίνεται η βέλτιστη τιμή z:

$$z^0 \leq z \leq z_{LP} \tag{4.13}$$

Η (4.13), εκτός από το διάστημα τιμών για τη z, μας δίνει και ένα άνω φράγμα για το πόσο "κακή" (υποβέλτιστη) είναι η λύση $\underline{x}^0$ που έχουμε βρεί. Πραγματικά από την (4.13) προκύπτει

$$0 \leq z - z^0 \leq z_{LP} - z^0 \tag{4.14}$$

που σημαίνει ότι το <u>χάσμα βελτιστότητας</u> (suboptimality gap) της x^0, $z - z^0$, είναι το πολύ $z_{LP} - z^0$. Επίσης, αν επιπλέον ισχύει $z^0 \geq 0$, τότε

$$\frac{z - z_0}{z} \leq \frac{z_{LP} - z^0}{z} \leq \frac{z_{LP} - z^0}{z^0} \tag{4.15}$$

Η (4.15) δίνει ένα άνω φράγμα για το ποσοστό υποβελτιστότητας (δηλαδή το σχετικό λάθος) της x^0, αν τη δεχθούμε ως προσεγγιστική λύση του προβλήματος (4.10). Είναι σημαντικό να τονίσουμε πάλι ότι στην παραπάνω συζήτηση η z (και επομένως και η βέλτιστη λύση $\underline{x}^*$ της (4.10)) μπορεί να είναι πολύ δύσκολο να υπολογιστούν επακριβώς. Αντίθετα η z_{LP} προκύπτει ως λύση ενός Π.Γ.Π, ενώ η x^0 και η z^0 μπορεί να προκύπτουν από την εφαρμογή ενός προσεγγιστικού αλγορίθμου για τη λύση του π.α.π. Τέτοιας μορφής αλγόριθμοι προσέγγισης χρησιμοποιούνται συχνά σε προβλήματα ακέραιου προγραμματισμού. Η ποιότητά τους αξιολογείται από το συνδυασμό της ταχύτητας εύρεσης της λύσης $\underline{x}^0$ όπως επίσης και του πόσο κοντά βρίσκεται η λύση στη βέλτιστη, δηλαδή από ένα σχετικό σφάλμα της μορφής (4.15).

Πρέπει εδώ να σημειώσουμε ότι σε πολλά προβλήματα α.π. η χαλάρωση γ.π. έχει μεγάλη διαφορά από το αρχικό πρόβλημα. Αυτό μπορεί να έχει ως συνέπεια να βρεθεί μια προσεγγιστική λύση x^0 για την οποία η υποβελτιστότητα $z-z^0$ να είναι στην πραγματικότητα μικρή αλλά επειδή $z_{LP}-z >> z-z^0$, η (4.14) και η (4.15) να δίνουν κακές εκτιμήσεις για την ποιότητα της x^0. Για το λόγο αυτό είναι χρήσιμο να βρεθούν καλύτερα άνω φράγματα για τη z απ' ότι αυτό της χαλάρωσης γ.π. Ένα μεγάλο μέρος της έρευνας στον α.π. εστιάζεται στην εύρεση τέτοιων φραγμάτων για διάφορες κατηγορίες προβλημάτων.

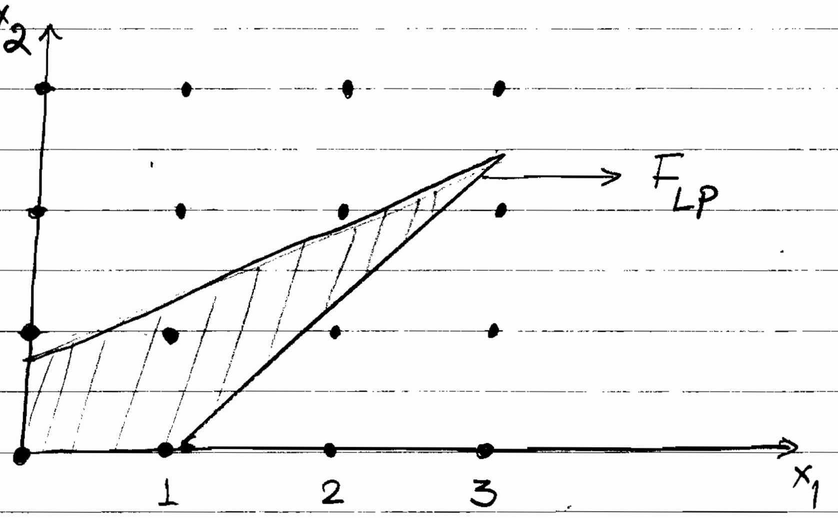
<u>Παράδειγμα 4.1.</u> Ας θεωρήσουμε το π.α.π.

$$
\begin{aligned}
z = \max\ & x_1 + x_2 \\
& -17x_1 + 30x_2 \le 24 \\
& 25x_1 - 18x_2 \le 30 \\
& x_1, x_2 \ge 0,\ x_1, x_2 \in \mathbb{Z}
\end{aligned}
$$

Η χαλάρωση γ.π. είναι $\quad z_{LP} = \max x_1 + x_2$

$$-17x_1 + 30x_2 \le 24$$
$$25x_1 - 18x_2 \le 30$$
$$x_1, x_2 \ge 0$$

Στο Σχήμα 4.4. φαίνεται η εφικτή περιοχή F_{LP} του z_{LP} και οι εφικτές (ακέραιες) λύσεις του π.α.π.



Σχήμα 4.4

Το σύνολο των εφικτών λύσεων του π.α.π. είναι $F = \{(0,0), (1,0), (1,1)\}$
Επομένως η βέλτιστη λύση είναι $x^* = (1,1)$ με $z = 2$
Από την άλλη πλευρά η χαλάρωση γ.π. έχει λύση
$z_{LP} = \frac{11}{2}$ με $x^*_{LP} = \left(3, \frac{5}{2}\right)$

Αν ένας προσεγγιστικός αλγόριθμος μας δώσει ως λύση του z την

$x^0 = (1,0)$ τότε $z^0 = 1$. Χωρίς να γνωρίζουμε την τιμή του z θα λέγαμε από την (4.15) ότι $\frac{z - z_0}{z} \le \frac{z_{LP} - z^0}{z^0} = 4.5$

δηλαδή ότι το χάσμα βελτιστότητας της z^0 είναι το πολύ 4.5 φορές η τιμή του z, ενώ στην πραγματικότητα ισχύει
$\frac{z - z_0}{z} = 1$ δηλ. 1 φορά.

4.3.2. Η μέθοδος κλάδου-φράγματος

Θεωρούμε πάλι το πρόβλημα (4.10), και έστω F η εφικτή περιοχή

$F=\{\underline{x}\in \mathbb{Z}^n \mid A\underline{x}=b,\ \underline{x}\geq 0\}$, οπότε $z=\max\{\underline{c}'\underline{x} : \underline{x}\in F\}$

Ας θεωρήσουμε μια διαμέριση του συνόλου $F = F_1 \cup F_2$ όπου $F_1 \cap F_2 = \phi$. Τότε ένας τρόπος να λύσουμε το πρόβλημα (4.10) είναι να λύσουμε τα δύο υποπροβλήματα

$$z(F_1) = \max\{\underline{c}'\underline{x} : \underline{x}\in F_1\}, \qquad z(F_2) = \max\{\underline{c}'\underline{x} : \underline{x}\in F_2\}$$

και να πάρουμε την καλύτερη από τις δύο λύσεις, δηλαδή

$$z(F) = z = \max\{z(F_1), z(F_2)\} \tag{4.16}$$

Τα προβλήματα $z(F_1)$, $z(F_2)$ είναι επίσης π.α.π, επομένως αν εφαρμόσουμε χαλάρωση γ.π. σε καθένα απ' αυτά παίρνουμε δύο άνω φράγματα:

$$z(F_1) \leq z_{LP}(F_1) \tag{4.17}$$
$$z(F_2) \leq z_{LP}(F_2)$$

Ας υποθέσουμε επίσης ότι έχουμε με κάποιο τρόπο βρεί μια εφικτή λύση $x^0 \in F$ του αρχικού προβλήματος, με $z^0 = \underline{c}'\underline{x}^0$ την τιμή της αντικ. συνάρτησης, επομένως $z^0 \leq z$.

Η βασική ιδέα της μεθόδου κλάδου-φράγματος είναι η εξής:
Ας υποθέσουμε ότι για τη λύση $\underline{x}^0$ ισχύει

$$z^0 \geq z_{LP}(F_1) \tag{4.18}$$

δηλ. η $\underline{x}^0$ είναι καλύτερη από τη βέλτιστη λύση της χαλάρωσης για το F_1. Από τις (4.16)-(4.18) προκύπτει ότι

$$z(F_1) \leq z_{LP}(F_1) \leq z^0 \leq z \tag{4.19}$$

Από την (4.19) βλέπουμε ότι δεν έχει νόημα να λύσουμε το πρόβλημα $z(F_1)$ ακριβώς, γιατί η λύση του αποκλείεται να είναι καλύτερη από την ήδη υπάρχουσα λύση $\underline{x}^0$. Επομένως αρκεί να λύσουμε το πρόβλημα $z(F_2)$ και θα έχουμε

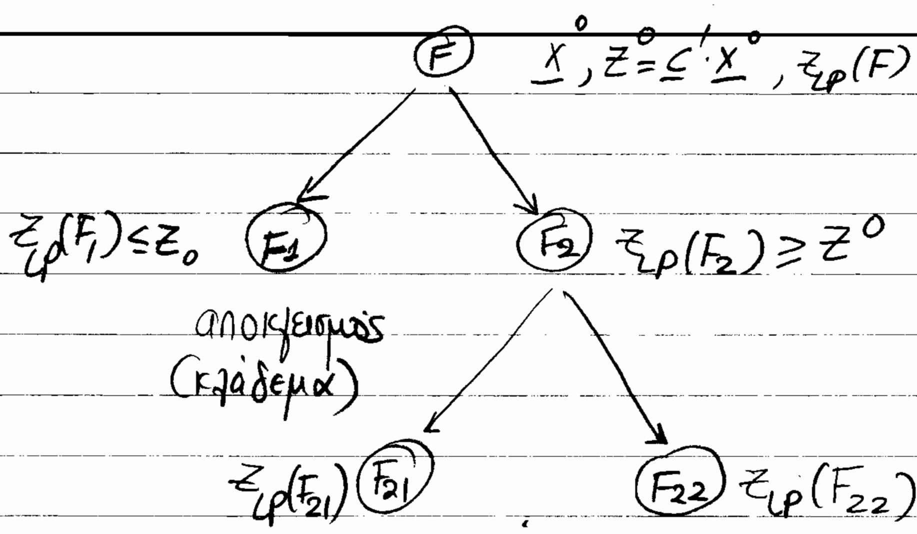
$$z = \max\{z^0, z(F_2)\}$$

Τώρα το πρόβλημα $z(F_2)$ είναι π.α.π. Επομένως μπορούμε να εφαρμόσουμε αναδρομικά τον παραπάνω συλλογισμό, δηλαδή να σχηματίσουμε μια διαμέριση του $F_2 = F_{21} \cup F_{22}$ και να συνεχίσουμε με τον "ίδιο τρόπο".

Από την άλλη πλευρά, αν δεν ισχύει καμμιά ανισότητα της μορφής (4.18) δηλαδή αν $z^0 \le z_{LP}(F_1)$, $z^0 \le z_{LP}(F_2)$, τότε δεν μπορούμε να αποκλείσουμε κανένα από τα υποπροβλήματα. Σ' αυτή την περίπτωση θα πρέπει να προχωρήσουμε αναδρομικά και στα δύο υποπροβλήματα με διαμερίσεις.

Βλέπουμε ότι έχουν εμφανιστεί τα σπέρματα μιας επαναληπτικής διαδικασίας πού λύνει διαδοχικά μικρότερα υποπροβλήματα, ενώ σε κάποιο στάδιο ένα υποπρόβλημα μπορεί να αποκλειστεί, δηλαδή να μη διαμεριστεί περαιτέρω, επειδή αποκλείεται να δώσει βέλτιστη λύση. Η διαδικασία αυτή μπορεί να περιγραφεί σχηματικά με τη βοήθεια ενός γραφήματος δέντρου όπως στο Σχήμα 4.5. Στο γράφημα κάθε κόμβος αντιστοιχεί σε ένα υποπρόβλημα ακέραιου προγραμματισμού με την αντίστοιχη εφικτή περιοχή.

Αν ένα υποπρόβλημα αποκλειστεί λόγω ανισότητας του τύπου (4.18) τότε λέμε ότι έχει γίνει <u>κλάδεμα</u> (pruning) του δέντρου στον κόμβο αυτό, καθώς δεν προχωρούμε στην ανάπτυξη κλάδων από τον συγκεκριμένο κόμβο.



Σχήμα 4.5

Η περιγραφή του αλγορίθμου κλάδου-φράγματος δεν είναι ακόμη πλήρη καθώς δεν έχουμε καθορίσει πώς γίνεται η διαμέριση, ούτε πότε τελειώνει η διαδικασία. Επίσης δεν έχουμε πει τίποτα για το πώς βρίσκουμε την λύση $\underline{x}^0$. Αυτά τα ερωτήματα θα δούμε αμέσως.

Πρώτα απ' όλα αν σε κάποιο υποπρόβλημα $z(F_\alpha)$ βρούμε τη βέλτιστη λύση της χαλάρωσης γ.π. και ισχύει $x^*_{LP}(F_a) \in Z^n$, δηλ. είναι ακέραια λύση, τότε όπως έχουμε δει, ισχύει $z(F_a) = z_{LP}(F_a)$. Σ' αυτή την περίπτωση επίσης δεν έχει νόημα να προχωρήσουμε σε παραπέρα διαμέριση του F_α, αφού το έχουμε επιλύσει ακριβώς. Έχουμε δηλαδή πάλι "κλάδεμα" του δέντρου στον κόμβο F_a. Εδώ όμως έχουμε και κάτι παραπάνω. Αυτό είναι η λύση $x^*_{LP}(F_a)$, που, αφού είναι ακέραια, αποτελεί μια νέα εφικτή λύση του $z(F)$. Αν αυτή η νέα λύση προκύψει ότι είναι καλύτερη από την προηγούμενη $\underline{x}^0$, δηλαδή $\underline{c}'x^*_{LP}(F_a) > \underline{c}'\underline{x}^0 = z^0$, τότε μπορούμε να ξεχάσουμε τη $\underline{x}^0$ και για τα επόμενα βήματα να χρησιμοποιήσουμε τη νέα λύση $x^*_{LP}(F_a)$ ως λύση $\underline{x}^0$. Αυτό είναι σημαντικό χαρακτηριστικό του αλγορίθμου, επειδή έτσι κάθε φορά έχουμε στη διάθεσή μας την καλύτερη εφικτή λύση που έχει παραχθεί μέχρι το συγκεκριμένο κόμβο του δέντρου. Επίσης, επειδή η νέα λύση έχει μεγαλύτερη τιμή του z, είναι ευκολότερο να γίνουν κλαδέματα σε επόμενα βήματα, λόγω ανισοτήτων της μορφής (4.18). Κλάδεμα αυτού του τύπου ονομάζεται εξιχνίαση (fathoming) του υποπροβλήματος-κόμβου.

Ας υποθέσουμε τώρα ότι σε ένα κόμβο-υποπρόβλημα F_a η βέλτιστη λύση $x^*_{LP}(F_a) \notin \mathbb{Z}^n$, δηλαδή έχει τουλάχιστον μια μη ακέραιη συνιστώσα. Έστω χωρίς βλάβη της γενικότητας ότι στη λύση $x^*_{LP}(F_a)$ ισχύει $x^*_1 \notin \mathbb{Z}$. Τότε μια διαμέριση του συνόλου F_a είναι η εξής:

$$F_{a1} = \{ \underline{x} \in F_a,\ x_1 \le \lfloor x^*_1 \rfloor \},\quad F_{a2} = \{ \underline{x} \in F_a,\ x_1 \ge \lfloor x^*_1 \rfloor + 1 \}$$

όπου με $\lfloor x \rfloor$ συμβολίζουμε το ακέραιο μέρος του x.
Η ιδέα της διαμέρισης δηλαδή είναι να προσδιορίσουμε μια μη ακέραιη συνιστώσα της $x^*_{LP}(F_a)$ και να δημιουργήσουμε δύο υποπροβλήματα της παραπάνω μορφής. Με τον τρόπο αυτό σε κανένα από τα δύο υποπροβλήματα F_{a1}, F_{a2} η $x^*_{LP}(F_a)$ δεν είναι εφικτή λύση αφού η $x^*_1 \notin \mathbb{Z}$ και έχει αποκλειστεί και από τα δύο λόγω του ορισμού.

Βλέπουμε λοιπόν ότι η διαμέριση έχει σκοπό να αποκλείσει κάποιες μη ακέραιες λύσεις που προκύπτουν από τη χαλάρωση Γ.Π., χωρίς όμως να αποκλείει ακέραιες λύσεις, πράγμα που θα ήταν λάθος.
Παρατηρούμε επίσης ότι σε κάθε ένα από τα F_{a1}, F_{a2} έχει προστεθεί ένας γραμμικός περιορισμός, επομένως και τα δύο συνεχίζουν να είναι Π.Α.Π., δηλαδή μπορεί να συνεχίσει

η αναδρομική διαδικασία.

Το επόμενο ερώτημα είναι πότε σταματά ο αλγόριθμος και αν βρίσκει τη βέλτιστη λύση του αρχικού προβλήματος.

Ο αλγόριθμος σταματά όταν όλα τα υποπροβλήματα-κόμβοι έχουν κλαδευτεί (είτε λόγω φράγματος είτε λόγω επίλυσης). Τότε η καλύτερη λύση που έχει διατηρηθεί είναι και η βέλτιστη λύση του προβλήματος. Αυτό προκύπτει εύκολα από τους συλλογισμούς που αναπτύξαμε προηγουμένως για τους μηχανισμούς κλαδέματος και εξιχνίασης.

Παρακάτω δίνουμε μια συνοπτική περιγραφή του αλγορίθμου κλάδου-φράγματος. Σε κάθε βήμα υπάρχει ένα σύνολο από υποπροβλήματα (κόμβους) που δεν έχουν εξεταστεί. Αυτό ονομάζεται σύνολο ενεργών υποπροβλημάτων ή ενεργών κόμβων (active nodes). Επίσης μπορεί να υπάρχει μια εφικτή λύση του αρχικού προβλήματος $\underline{x}^0$ με τιμή $z^0 = \underline{c}'\cdot\underline{x}^0$ (αν έχει ήδη δημιουργηθεί). Αυτή ονομάζεται τρέχουσα λύση (incumbent solution).

Ο αλγόριθμος κλάδου-φράγματος λειτουργεί ως εξής:

1. Επιλέγουμε έναν από τους ενεργούς κόμβους (F_i)
2. Για το υποπρόβλημα του κόμβου υπολογίζουμε το φράγμα της χαλάρωσης γ.π. $z_{LP}(F_i)$
3. Αν ισχύει $z_{LP}(F_i) \leq z^0$ ο κόμβος κλαδεύεται και παύει να είναι ενεργός.
4. Αν $z_{LP} > z^0$ τότε:

   4a. Αν η λύση $x^*_{LP}(F_i) \in \mathbb{Z}^n$ τότε ο κόμβος έχει εξιχνιαστεί (fathomed). Αν $\underline{c}'x^*_{LP}(F_i) > z^0$, Τότε η $\underline{x}^*_{LP}(F_i)$ γίνεται η νέα τρέχουσα λύση

$$\underline{x}_0 \leftarrow \underline{x}^*_{LP}(F_i), \quad z_0 \leftarrow \underline{c}' \cdot \underline{x}^*_{LP}(F_i)$$

διαφορετικά η τρέχουσα λύση παραμένει η προηγούμενη
Ο κόμβος παύει να είναι ενεργός και στις δύο περιπτώσεις

(4β) Αν $\underline{x}^*_{LP}(F_i) \notin \mathbb{Z}_n$, επιλέγουμε μια μη ακέραια συνιστώσα της $\underline{x}^*_{LP}(F_i)$, έστω την $x_1^*(F_i)$

Δημιουργούμε δύο νέους ενεργούς κόμβους F_{i1}, F_{i2} που προκύπτουν από το υποπρόβλημα F_i με την προσθήκη των περιορισμών

$$x_1 \le \lfloor x_1^* \rfloor \quad \text{στο} \quad F_{i1}$$

$$x_1 \ge \lfloor x_1^* \rfloor + 1 \quad \text{στο} \quad F_{i2}$$

Ο κόμβος F_i παύει να είναι ενεργός.

Παρατηρήσεις (1) Αν σε ένα υποπρόβλημα η χαλάρωση γ.π. είναι μη εφικτό π.γ.π. ο κόμβος διαγράφεται.

(2) Ο αλγόριθμος σταματά όταν εξαντληθεί η λίστα ενεργών κόμβων. Τότε, αν δεν έχει δημιουργηθεί τρέχουσα λύση το πρόβλημα είναι ανέφικτο, διαφορετικά η τρέχουσα λύση είναι βέλτιστη

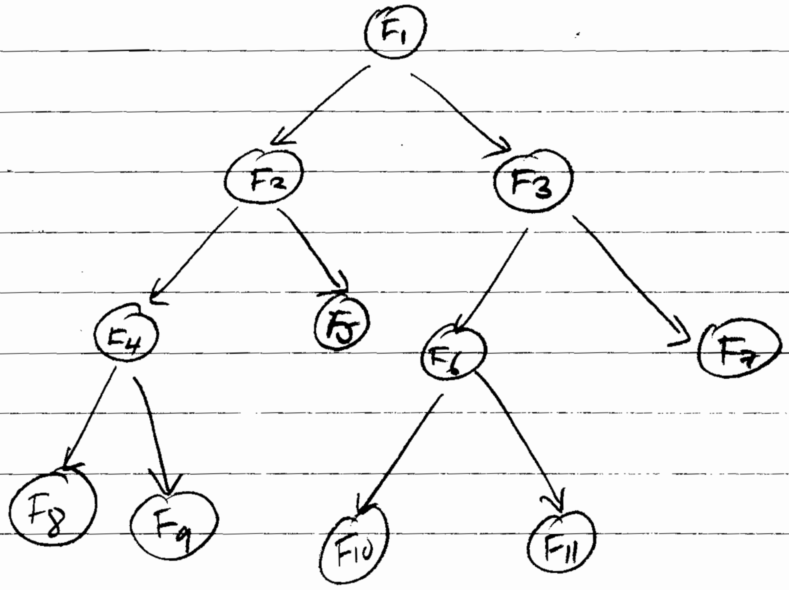
(3) Αν σε κάποιο βήμα υπάρχουν περισσότεροι από έναν ενεργοί κόμβοι, τότε υπάρχουν διάφορες εναλλακτικές στρατηγικές

για την επιλογή αυτού που θα εξεταστεί. Οι πιο συνηθισμένες είναι:

(a) κατά βάθος πρώτα (depth-first) Όταν δημιουργούνται δύο νέοι κλάδοι, δηλαδή το δέντρο προχωράει κατά ένα επίπεδο από αυτό τον κλάδο, πρώτα εξετάζεται ένας από τους νέους κόμβους, και αυτό συνεχίζεται αναδρομικά

(b) κατά εύρος πρώτα (breadth-first). Όταν τελειώσει η εξέταση ενός κόμβου, επιλέγεται ένας ενεργός κόμβος από το ίδιο επίπεδο (αν υπάρχει). Όταν τελειώσουν οι κόμβοι ενός επιπέδου προχωρούμε στην εξέταση κόμβων από το επόμενο επίπεδο.

Σχηματικά ας θεωρήσουμε ότι σε ένα πρόβλημα υπάρχουν τα υποπροβλήματα που φαίνονται στο Σχήμα 4.6.



Σχήμα 4.6

Ο αλγόριθμος κατά-βάθος-πρώτα θα τα εξετάσει με τη σειρά $F_1 \to F_2 \to F_4 \to F_8 \to F_9 \to F_5 \to F_3 \to F_6 \to F_{10} \to F_{11} \to F_7$
ενώ ο αλγόριθμος κατά εύρος πρώτα
με τη σειρά $F_1 \to F_2 \to F_3 \to F_4 \to F_5 \to F_6 \to F_7 \to F_8 \to F_9 \to F_{10} \to F_{11}$

④ Η χαλάρωση γ.π. δίνει ένα τρόπο για τον υπολογισμό άνω φράγματος σε κάθε κόμβο. Στη βιβλιογραφία έχουν προταθεί εναλλακτικοί τρόποι για τον υπολογισμό φραγμάτων (όπως π.χ. Λαγκρανζιανή χαλάρωση) που εκμεταλλεύονται τη συγκεκριμένη δομή του προβλήματος.

⑤ Προφανώς αν αντί για μεγιστοποίηση έχουμε πρόβλημα ελαχιστοποίησης μπορούμε να σχεδιάσουμε ένα ακριβώς αντίστοιχο αλγόριθμο (αν δε θέλουμε να μετατρέψουμε το πρόβλημα σε κανονική μορφή). Τώρα τα φράγματα χαλάρωσης θα είναι κάτω φράγματα και οι ανισότητες (4.17) - (4.19) θα ισχύουν με αντίστροφη φορά.

⑥ Για την περίπτωση 0-1 προγραμματισμού (όλες οι μεταβλητές 0-1) ο αλγόριθμος μπορεί να εφαρμοστεί με τον ίδιο τρόπο. Η μόνη διαφορά είναι ότι κατά τη διακλάδωση με μεταβλητή x_j, τα δύο υποπροβλήματα θα έχουν περιορισμούς $x_j=0$ και $x_j=1$ αντίστοιχα.

⑦ Για τη γενική περίπτωση μικτού προγραμματισμού μπορούμε να εφαρμόσουμε τον παραπάνω αλγόριθμο με τη διαφορά ότι διακλαδώσεις γίνονται μόνο για μεταβλητές που έχουν ακέραιους περιορισμούς.

Παράδειγμα 4.2 Να λυθεί το παρακάτω π.α.π. με τον αλγόριθμο κλάδου-φράγματος:

$$Z = \max \ 26x_1 + 28x_2 - 2000x_3$$

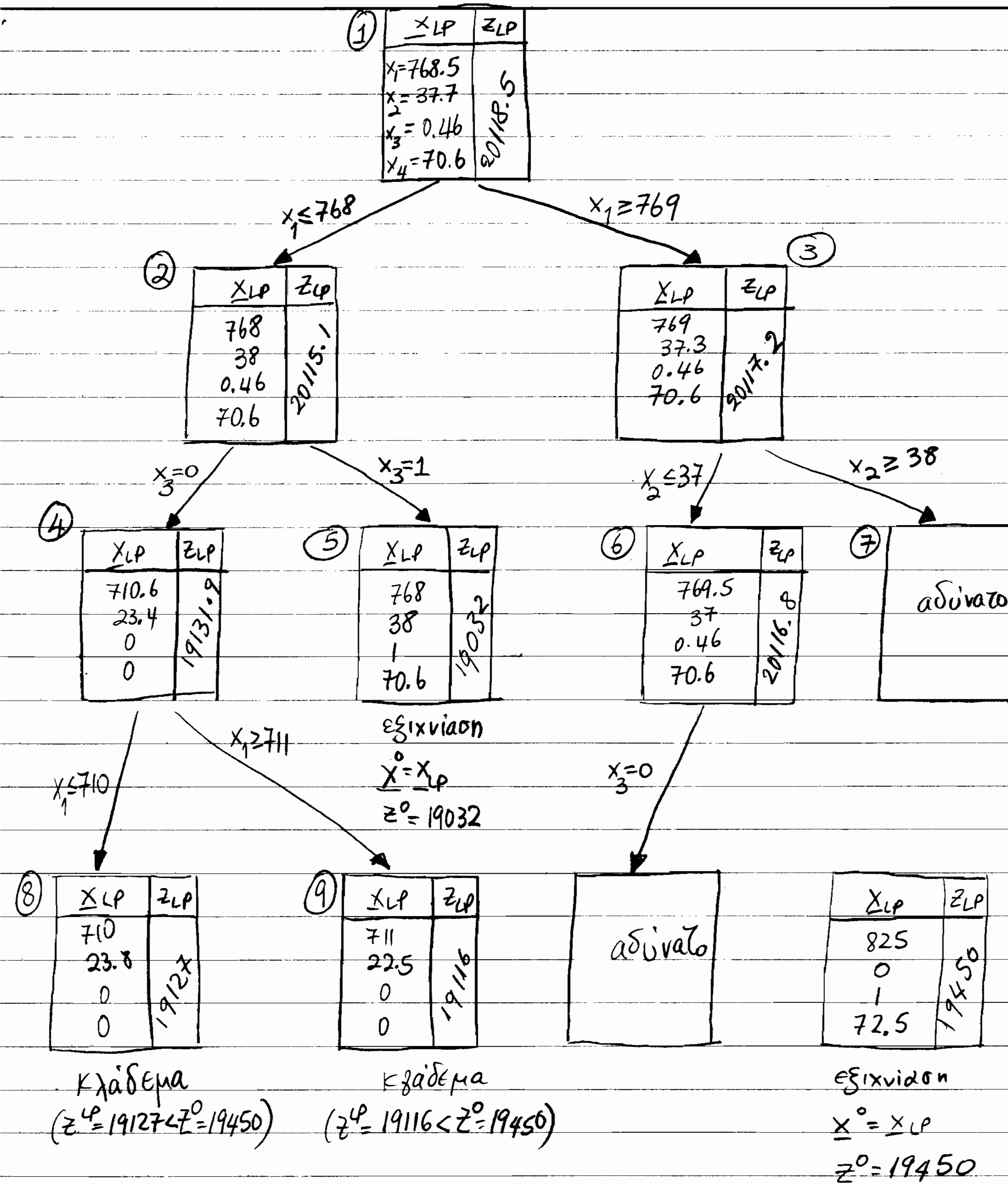
$$\begin{aligned}
\text{υ.π.} \quad & 9x_1 + 13x_2 - 10x_4 \le 6700 \\
& 7x_1 + 6x_2 \le 6200 \\
& 10x_1 + 4x_2 - 1540x_3 + x_4 \le 7200 \\
& 2x_1 + 3x_2 \le 1650 \\
& -154x_3 + x_4 \le 0 \\
& x_1, x_2, x_3, x_4, x_5 \ge 0 \\
& x_1, x_2 \in \mathbb{Z}, \quad x_3 \in \mathbb{B}
\end{aligned}$$

Στο πρόβλημα αυτό έχουμε 2 ακέραιες και μία 0-1 μεταβλητή. Για τη μεταβλητή x_4 πού δεν έχει περιορισμό ακέραιων τιμών δε θα γίνουν διακλαδώσεις.

Επίσης το πρόβλημα δεν είναι σε κανονική μορφή. Αυτό δε δημιουργεί δυσκολία, καθώς η μέθοδος βασίζεται σε λύση διαδοχικών προβλημάτων γραμμικού προγραμματισμού, τα οποία μπορούν να είναι ή όχι σε κανονική μορφή.

Η συνολική εικόνα του δέντρου που προκύπτει από την εφαρμογή της μεθόδου κλάδου φράγματος παρουσιάζεται στο Σχήμα 4.7. Σε κάθε κόμβο φαίνεται η λύση $\underline{x}_{LP}$ της χαλάρωσης γ.π. και το άνω φράγμα Z_{LP}.

Στους κλάδους φαίνεται ο περιορισμός που προστίθεται στο αντίστοιχο υποπρόβλημα.


1
$\underline{x}_{LP}$ z_{LP}
$x_1=768.5$
$x_2=37.7$
$x_3=0.46$
$x_4=70.6$
20118.5
$x_1 \le 768$
$x_1 \ge 769$
2
768
38
0.46
70.6
20115.1
3
769
37.3
0.46
70.6
20117.2
$x_3=0$
$x_3=1$
$x_2 \le 37$
$x_2 \ge 38$
4
710.6
23.4
0
0
19131.9
5
768
38
1
70.6
19032
εξιχνίαση
$\underline{x}^0=\underline{x}_{LP}$
$z^0=19032$
6
769.5
37
0.46
70.6
20116.8
7
αδύνατο
$x_1 \le 710$
$x_1 \ge 711$
$x_3=0$
8
710
23.8
0
0
19127
κλάδεμα
$(z^{LP}=19127<z^0=19450)$
9
711
22.5
0
0
19116
κλάδεμα
$(z^{LP}=19116<z^0=19450)$
αδύνατο
825
0
1
72.5
19450
εξιχνίαση
$\underline{x}^0=\underline{x}_{LP}$
$z^0=19450$

Σχήμα 4.7

Ακολουθούμε τη στρατηγική εξέτασης κατά εύρος πρώτα.

Στον κόμβο 1 λύνουμε τη χαλάρωση γ.π. του αρχικού προβλήματος

$$
\begin{aligned}
Z_{LP}(F_1) = \max\ & 26x_1 + 28x_2 - 2000x_3 \\
& 9x_1 + 13x_2 \qquad\qquad -10x_4 \le 6700 \\
& 7x_1 + 6x_2 \qquad\qquad\qquad \le 6200 \\
& 10x_1 + 4x_2 - 1540x_3 + x_4 \le 7200 \\
& 2x_1 + 3x_2 \qquad\qquad\qquad \le 1650 \\
& \qquad -154x_3 + x_4 \le 0 \\
& \qquad\qquad x_3 \le 1 \\
& x_1, x_2, x_3, x_4 \ge 0
\end{aligned}
$$

(επειδή αρχικά $x_3 \in \{0,1\}$)

Η λύση του $Z_{LP}(F_1)$ είναι $\underline{X}_{LP}(F_1) = \begin{pmatrix} 768.5 \\ 37.7 \\ 0.46 \\ 70.6 \end{pmatrix}$, και $Z_{LP} = 20118.5$

Επειδή δεν ικανοποιείται η ακεραιότητα των x_1, x_2, x_3 κάνουμε διακλάδωση με βάση τη μεταβλητή x_1 (θα μπορούσαμε να επιλέξουμε και τη x_2 ή τη x_3). Έτσι δημιουργούμε δύο νέους κόμβους. Στον κόμβο 2 επιλύουμε το π.γ.π. που προκύπτει από το $Z_{LP}(F_1)$ με την προσθήκη του περιορισμού $x_1 \le 768$, ενώ στον κόμβο 3 το π.γ.π. που προκύπτει από το $Z_{LP}(F_1)$ με την προσθήκη του περιορισμού $x_1 \ge 769$. Επειδή η στρατηγική είναι κατά εύρος πρώτα, επιλύουμε και τα δύο προβλήματα F_2, F_3 πριν προχωρήσουμε σε τυχόν περαιτέρω διακλαδώσεις

Βρίσκουμε $Z_{LP}(F_2) = 20115.1$, $\underline{X}_{LP}(F_2)$ μη ακέραιη
$Z_{LP}(F_3) = 20117.3$, $\underline{X}_{LP}(F_3)$ μη ακέραιη

Σ' αυτό το σημείο οι ενεργοί κόμβοι είναι οι 2 και 3.

Στον κόμβο 2: Διακλαδώνουμε με βάση τη x_3 και δημιουργούμε τον κόμβο 4 με την προσθήκη του περιορισμού $x_3=0$ στο υποπρόβλημα F_2, και τον κόμβο 5 με την προσθήκη του περιορισμού $x_3=1$ στο υποπρόβλημα F_2.

Στον κόμβο 3: Διακλαδώνουμε με βάση την x_2, και δημιουργούμε τους κόμβους 6 και 7, προσθέτοντας τους περιορισμούς $x_2 \le 37$ και $x_2 \ge 38$, αντίστοιχα, στο υποπρόβλημα F_3.

Τώρα το σύνολο ενεργών κόμβων είναι $\{4,5,6,7\}$

Λύνουμε τα υποπροβλήματα και βρίσκουμε

$z_{LP}(F_4) = 19131.9$, $\underline{x}_{LP}(F_4)$ μη ακέραιη

$z_{LP}(F_5) = 19032$, $\underline{x}_{LP}(F_5) = \begin{pmatrix} 768 \\ 38 \\ 1 \\ 70.6 \end{pmatrix}$

Επειδή η $\underline{x}_{LP}(F_5)$ έχει ακέραιες τιμές x_1, x_2, x_3 αποτελεί την πρώτη τρέχουσα λύση:

$$\underline{x}^0 = \begin{pmatrix} 768 \\ 38 \\ 1 \\ 70.6 \end{pmatrix}, \quad z^0 = 19032$$

Ο κόμβος 5 έχει εξιχνιαστεί και παύει να είναι ενεργός, ούτε κάνουμε διακλαδώσεις

$z_{LP}(F_6) = 20116.8$, $\underline{x}_{LP}(F_6)$ μη ακέραιη

Επειδή $z_{LP}(F_6) > z^0$, δεν μπορεί να γίνει κλάδεμα στον κόμβο 6. Υπάρχει περιθώριο με περαιτέρω διακλαδώσεις από τον κόμβο 6 να καταλήξουμε σε ακέραιη λύση καλύτερη από την τρέχουσα.

$Z_{LP}(F_7)$: ανέφικτο πρόβλημα, επομένως στον κόμβο 7 γίνεται κλάδεμα.

Επιστρέφουμε στον κόμβο 4: Επειδή και εδώ $Z_{LP}(F_4) > Z^0$ δεν μπορούμε να κάνουμε κλάδεμα.
Κάνουμε διακλάδωση με βάση τη μεταβλητή $x_1 = 710.6$ και δημιουργούμε τους κόμβους 8, 9 με την προσθήκη των περιορισμών $x_1 \leq 710$ και $x_1 \geq 711$, αντίστοιχα στο υποπρόβλημα F_4.

Στον κόμβο 6 κάνουμε διακλάδωση με βάση τη μεταβλητή x_3 και δημιουργούμε τους κόμβους 10, 11 με την προσθήκη των περιορισμών $x_3 = 0$ και $x_3 = 1$, αντίστοιχα, στο υποπρόβλημα F_6.

Βρίσκουμε

$Z_{LP}(F_8) = 19127$, $\underline{x}_{LP}(F_8)$ μη ακέραιη

$Z_{LP}(F_9) = 19116$, $\underline{x}_{LP}(F_9)$ μη ακέραιη

$Z_{LP}(F_{10})$ αδύνατο $\Rightarrow$ κλάδεμα

$Z_{LP}(F_{11}) = 19450$, $\underline{x}_{LP}$ ακέραιη,

Επομένως ο κόμβος 11 έχει εξιχνιαστεί.
Η νέα λύση έχει $Z_{LP}(F_{11}) = 19450 > Z^0 = 19032$

Επομένως γίνεται αυτή η τρέχουσα λύση:

$$\underline{x}^0 = \begin{pmatrix} 825 \\ 0 \\ 1 \\ 72.5 \end{pmatrix} \qquad z^0 = 19450$$

και ο κόμβος 11 κλαδεύεται.

Οι ενεργοί κόμβοι τώρα είναι οι 8 και 9.

Επειδή $z_{LP}(F_8) < z^0$, ο 8 κλαδεύεται.

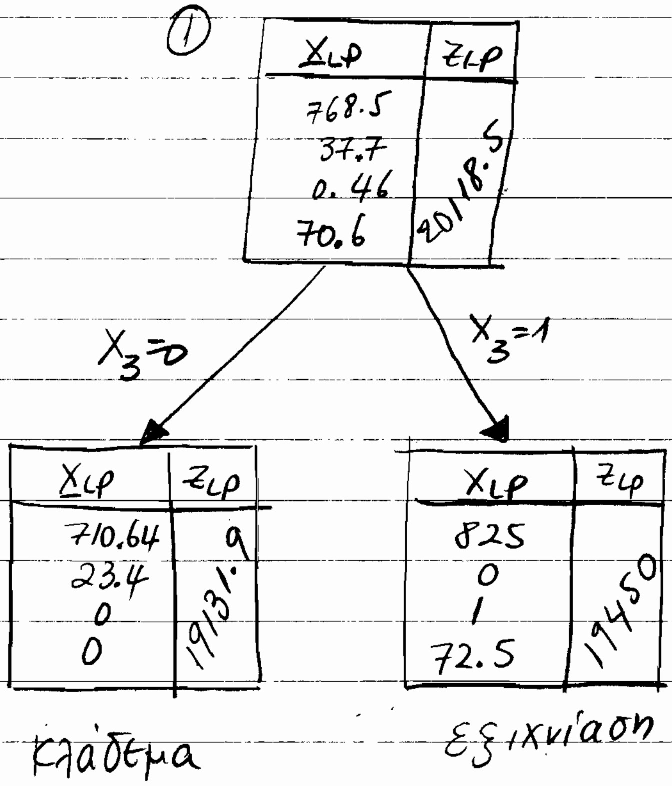
Επειδή $z_{LP}(F_9) < z^0$, ο 9 κλαδεύεται.

Τώρα έχουν εξαντληθεί οι ενεργοί κόμβοι και ο αλγόριθμος σταματά. Η τρέχουσα λύση $\underline{x}^0$ είναι η βέλτιστη λύση του αρχικού προβλήματος:

$$\underline{x}^* = \begin{pmatrix} 825 \\ 0 \\ 1 \\ 72.5 \end{pmatrix} \qquad z^* = 19450$$

Πριν αφήσουμε το παράδειγμα σημειώνουμε ότι η στρατηγική εξέτασης κατά εύρος πρώτα και οι συγκεκριμένες επιλογές μεταβλητών για διακλάδωση σε κάθε κόμβο ήταν αυθαίρετες. Αν επιλέγαμε κάποια άλλη στρατηγική θα καταλήγαμε στην ίδια βέλτιστη λύση z^* φυσικά, αλλά με διαφορετικό δέντρο και αριθμό βημάτων. Για παράδειγμα αν στον κόμβο 1 επιλέγαμε τη μεταβλητή x_3 για διακλάδωση θα καταλήξαμε στο δέντρο του Σχήματος 4.8, όπου βρίσκουμε τη βέλτιστη λύση εξετάζοντας 3 μόνο κόμβους.

Βλέπουμε επομένως ότι ο τρόπος εξέτασης και διακλάδωσης μπορεί να επηρεάσει σημαντικά το χρόνο επίλυσης του προβλήματος. Δυστυχώς δεν υπάρχει καμμιά στρατηγική που να είναι η καλύτερη για όλες τις περιπτώσεις. Έχουν γίνει όμως πολυάριθμες μελέτες και ερευνητικές προσπάθειες για την αξιολόγηση των διαφόρων στρατηγικών σε ειδικές κατηγορίες προβλημάτων.



$$X^0 = \begin{pmatrix} 825 \\ 0 \\ 1 \\ 72.5 \end{pmatrix} \qquad z^0 = 19450$$

Σχήμα 4.8

4.4. ΑΣΚΗΣΕΙΣ

Άσκηση 4.1 Μια εταιρεία πρέπει να κάνει παράδοση παραγγελιών σε 10 πελάτες. Η ποσότητα που έχει παραγγείλει κάθε πελάτης είναι d_j, $j=1,\dots,10$. Η εταιρεία διαθέτει 4 φορτηγά. Το φορτηγό k έχει χωρητικότητα L_k και ημερήσιο κόστος C_k (αν χρησιμοποιηθεί), $k=1,\dots,4$. Ολόκληρη η παραγγελία κάθε πελάτη πρέπει να παραδοθεί από ένα μόνο φορτηγό. Κάθε φορτηγό μπορεί να κάνει το πολύ 5 παραδόσεις παραγγελιών. Επίσης τα ζεύγη πελατών $\{1,7\}$, $\{2,6\}$ και $\{2,9\}$ δεν μπορούν να εξυπηρετηθούν από το ίδιο φορτηγό. Δημιουργήστε ένα μοντέλο Α.Π. για την ελαχιστοποίηση του συνολικού κόστους των φορτηγών.

Άσκηση 4.2 Μια εταιρεία έχει δύο προϊόντα $k=1,2$, ένα εργοστάσιο, δύο κέντρα διανομής $i=1,2$ και πέντε μεγάλους πελάτες $j=1,\dots,5$, των οποίων η ζήτηση d_{jk} είναι γνωστή για κάθε ένα από τα δύο προϊόντα. Η εταιρεία μπορεί να διακινήσει κάθε προϊόν από ένα ή δύο κέντρα διανομής. Κάθε πελάτης πρέπει να πάρει όλη την ποσότητα κάθε προϊόντος από ένα μόνο κέντρο διανομής (όχι απαραίτητα το ίδιο και για τα δύο προϊόντα, π.χ. ο πελάτης 1 μπορεί να παραλάβει το προϊόν 1 από το κέντρο 1 και το προϊόν 2 από το κέντρο 2). Τα κόστη είναι:

f_{ik} = σταθερό κόστος αν το προϊόν k διακινείται στο κέντρο i

f_{ijk} = σταθερό κόστος αν η ζήτηση του πελάτη j για το προϊόν k ικανοποιείται από το κέντρο i

c_{ijk} = μοναδιαίο κόστος μεταφοράς προϊόντος k από κέντρο i στον πελάτη j

α) Να μοντελοποιηθεί το πρόβλημα διανομής για ελαχιστοποίηση του συνολικού κόστους.

β) Πώς αλλάζει το μοντέλο στο (α) αν η ζήτηση του πελάτη για ένα προϊόν μπορεί να διαμοιραστεί στα κέντρα διανομής;

Άσκηση 4.3. Θεωρούμε το πρόβλημα παραγωγής ενός προϊόντος σε ορίζοντα T περιόδων. Αν αποφασίσουμε να παράγουμε κατά την περίοδο t, υπάρχει ένα σταθερό κόστος ίσο με K_t ανεξάρτητα από την ποσότητα παραγωγής όπως επίσης και μεταβλητό κόστος ίσο με c_t ανά μονάδα παραγόμενου προϊόντος. Το κόστος αποθήκευσης κατά την περίοδο t είναι ίσο με h_t, $t=1,2,\dots,T$ (όπως στο κεφάλαιο 1). Η ζήτηση είναι ίση με d_t, $t=1,\dots,T$.

(α) Να μοντελοποιηθεί το πρόβλημα παραγωγής για ελαχιστοποίηση του συνολικού κόστους ως πρόβλημα μ.α.π.

(β) Έστω ότι η παραγωγή μπορεί να γίνει το πολύ σε πέντε περιόδους, αλλά χωρίς κάποιες απ' αυτές να είναι διαδοχικές. Πώς αλλάζει το μοντέλο στο (α) με τους νέους περιορισμούς;

Άσκηση 4.4 Θέλετε να κάνετε μετακόμιση. Έχετε n αντικείμενα μεγέθους a_j, $j=1,\dots,n$, που θα μετακινηθούν στο νέο χώρο. Η μετακόμιση θα γίνει με ένα φορτηγό χωρητικότητας Q. Κάθε αντικείμενο πρέπει να μπει σε ένα κουτί. Έχετε αγοράσει m κουτιά μεγέθους b_i, $i=1,\dots,m$.
Ο ιδιοκτήτης του φορτηγού χρεώνει σταθερή τιμή για κάθε κουτί που τοποθετείται στο φορτηγό ανεξάρτητα από το μέγεθος.

(α) Να μοντελοποιηθεί ένα πρόβλημα α.π. για την ελαχιστοποίηση του κόστους.

β) Τι ρόλο παίζει η παράμετρος Q;

Άσκηση 4.5 Έστω n εργασίες $j=1,\dots,n$, κάθε μία από τις οποίες πρέπει να εκτελεστεί σε μια μοναδική μηχανή. Κάθε εργασία απαιτεί χρόνο επεξεργασίας μιας μέρας. Οι εργασίες θα γίνουν διαδοχικά η μια μετά την άλλη και το πρόβλημα είναι να βρεθεί η σειρά εκτέλεσης. Κάθε εργασία j έχει κόστος c_j, $j=1,\dots,n$, ανά ημέρα καθυστέρησης, δηλαδή αν τελειώσει την ημέρα t θα υπάρχει κόστος ίσο με tc_j. Επίσης υπάρχουν κάποιοι περιορισμοί προτεραιότητας. Συγκεκριμένα η εργασία 3 δεν μπορεί να γίνει πριν από την εργασία 1, ενώ η εργασία 5 δεν μπορεί να γίνει πριν από την εργασία 2. Να μοντελοποιηθεί το πρόβλημα προγραμματισμού των εργασιών για ελαχιστοποίηση του συνολικού κόστους, ως πρόβλημα α.π.
(Υπόδειξη: Παραλλαγή του προβλήματος ανάθεσης)

Άσκηση 4.6 Έστω ένα πρόβλημα βελτιστοποίησης με εφικτή περιοχή

$$F = \Big\{ (x_1,x_2)\in \mathbb{R}^2 : \quad x_1, x_2 \geq 0, \quad \text{και} \quad (3x_1+2x_2 \leq 7 \ \text{ή}\ 2x_1+3x_2 \leq 7) \quad \text{και} \quad (2x_1+x_2 \geq 1 \ \text{ή}\ x_1+2x_2 \geq 1) \Big\}$$

Να εκφραστεί η F ως σύζευξη περιορισμών.

Άσκηση 4.7 Έστω ένα π.α.π.

$$\max c'x$$
$$Ax = b$$
$$0 \le x_j \le u_j \quad , j = 1, \dots, n$$
$$x_j \in \mathbb{Z} \quad , j = 1, \dots, n$$

Να κατασκευαστεί ένα ισοδύναμο πρόβλημα 0-1 προγραμματισμού.

Άσκηση 4.8 Έστω το π.α.π

$$\max \quad x_1 + 2x_2$$
$$-3x_1 + 4x_2 \le 4$$
$$3x_1 + 2x_2 \le 11$$
$$2x_1 - x_2 \le 5$$
$$x_1, x_2 \ge 0$$
$$x_1, x_2 \in \mathbb{Z}.$$

(α) Να λυθεί γραφικά

(β) Να λυθεί με τη μέθοδο κλάδου φράγματος, όπου τα π.γ.π. για κάθε υποπρόβλημα επιλύονται γραφικά.

Άσκηση 4.9 Να λυθεί με τη μέθοδο κλάδου φράγματος το παρακάτω π.α.π., χρησιμοποιώντας στρατηγική κατά εύρος πρώτα

max x_4

$$2x_1+2x_2+2x_3+x_4=3$$

$$x_j \in \{0,1\},\ j=1,2,3,4$$

Άσκηση 4.10 Να λυθεί το πρόβλημα της Άσκησης 4.9 με στρατηγική κατά βάθος πρώτα.

Βιβλιογραφία

1. Μηλολιδάκης, Κ. (2003) "*Βασική Θεωρία Βελτιστοποίησης (Σημειώσεις για το Μάθημα Γραμμικός και Μη Γραμμικός Προγραμματισμός)*".

2. Οικονόμου, Α., Φακίνος, Δ. (2002) *"Εισαγωγή στην Επιχειρησιακή Έρευνα"*, Εκδόσεις Συμμετρία, Αθήνα.

3. Bertsimas, D. and Tsitsiklis, J. (1997) *"Introduction to Linear Optimization"*, Athena Scientific, Boston.

4. Nemhauser, G. and Wolsey, (1988) *"Integer and Combinatorial Optimization"*, John Wiley, New York.

5. Padberg, W. (1999) *"Linear Optimization and Extensions"*, Springer, Berlin.

6. Murty, K. (1976) *"Linear and Combinatorial Programming"*, John Wiley, New York.