ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΜΕ ΘΕΜΑ:

ΕΠΙΧΕΙΡΗΣΙΑΚΗ ΕΡΕΥΝΑ: ΜΗ ΓΡΑΜΜΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

(Τετραγωνικός, Διαχωρίσιμος, Κυρτός, Μη κυρτός)

Της προπτυχιακής φοιτήτριας:

Σοφίας Κ. Ζαβόλα, Α.Μ. 09108033

Κατεύθυνση Μαθηματικού Εφαρμογών

Ροές: Ανάλυση – Εφαρμοσμένη Μηχανική

Επιβλέπων καθηγητής: Ιωάννης Κολέτσος, Επίκουρος Καθηγητής

Τριμελής Επιτροπή:

Κοκκίνης Βασίλειος, Επίκουρος Καθηγητής

Κολέτσος Ιωάννης, Επίκουρος Καθηγητής

Τυχόπουλος Ευάγγελος, Επίκουρος Καθηγητής

ΑΘΗΝΑ, ΙΟΥΝΙΟΣ 2015

Αφιερώνεται στην οικογένειά μου

Περιεχόμενα

Πρόλογος

Η Επιχειρησιακή Έρευνα (Operational Research) είναι ο κλάδος της επιστήμης (σύνθετος τομέας ειδικοτήτων) με αντικείμενο την εύρεση λύσεων για την λήψη αποφάσεων. Οι λύσεις αυτές είναι κατά κανόνα ‘‘βέλτιστες’’, ως προς τα δεδομένα του προβλήματος. Απλά παραδείγματα Ε.Ε. είναι η βέλτιστη κατανομή πόρων και οι βέλτιστες μεταφορές. Για παράδειγμα επιλογή τροφών για σίτιση στρατού, με ελάχιστο κόστος. Επίσης, η ελάχιστη διαδρομή μεταφοράς καυσίμων από το διυλιστήριο σε πρατήρια βενζίνης. Μία πολύ πιο δύσκολη εφαρμογή είναι ο προγραμματισμός (scheduling) σε πραγματικό χρόνο, σε πύργο ελέγχου πολυσύχναστου αεροδρομίου, με στόχο την ασφάλεια και ελαχιστοποίηση του μέσου χρόνου αναμονής αεροσκαφών.

Το βέλτιστο της λύσης ορίζεται ως προς κάποια μετρήσιμα κριτήρια, π.χ. ελάχιστο κόστος, μέγιστη ευστάθεια, ελάχιστος χρόνος αναμονής ή διεκπεραίωσης, μέγιστο όριο αντοχής, ελαχιστοποίηση κάποιας νόρμας, συνδυασμός κέρδους με ρίσκο, κ.τ.λ. Συχνά στις εφαρμογές, τα δεδομένα του προβλήματος είναι ελλιπή και οι ‘‘βέλτιστες’’ αποφάσεις λαμβάνονται υπό συνθήκη αβεβαιότητας. Για τις περιπτώσεις αυτές, χρησιμοποιούμε τη μεθοδολογία των Πιθανοτήτων και της Στατιστικής, με τους αντίστοιχους δείκτες αξιοπιστίας.

Η χαρακτηριστική διαδικασία επίλυσης ενός προβλήματος της Ε.Ε. είναι:

α) Η μαθηματική μοντελοποίηση του προβλήματος.

β) Η αξιοποίηση της δομής του μαθηματικού μοντέλου, για επινόηση κατάλληλων αλγορίθμων βελτιστοποίησης.

γ) Η αριθμητική λύση, με τη χρήση υπολογιστή.

Τα προβλήματα που τίθενται καλύπτουν ένα ευρύ φάσμα δραστηριοτήτων, με αποτέλεσμα η μοντελοποίηση ή προσομοίωση να αξιοποιεί διαφορετικούς επιστημονικούς κλάδους ή συνδυασμούς αυτών. Μία μερική εποπτεία αυτών των κλάδων που χρησιμοποιούνται ως ‘‘εργαλεία’’ της Ε.Ε. είναι: Στατιστική, Βελτιστοποίηση, Βέλτιστος Έλεγχος, Θεωρία Παιγνίων, Νευρωνικά Δίκτυα, Ρομποτική, Επιστήμη των Υπολογιστών, Μηχανική όλων των ειδικοτήτων, Management, Έλεγχος Αποθεμάτων (Inventory Control), ακόμα και στοιχεία Ψυχολογίας και Κοινωνιολογίας.

Ιστορικά, πολύ ενδιαφέρουσα είναι η ραγδαία εξέλιξη του κλάδου της Ε.Ε., ιδιαίτερα από τα πρώτα χρόνια του 2ου Παγκοσμίου Πολέμου μέχρι σήμερα. Αυτή η εξέλιξη θα μπορούσε ίσως να συγκριθεί με την εκθετική ανάπτυξη των υπολογιστών! Από την αρχή του πολέμου, οι Άγγλοι, οι Σοβιετικοί και οι Αμερικάνοι, επινόησαν μαθηματικούς τρόπους για ‘‘βέλτιστες αποφάσεις’’ σχετικά με πρόβλεψη κινήσεων εχθρικών υποβρυχίων, διαχείρισης πληροφοριών από ραντάρ, βέλτιστης κατανομής πόρων κ.ά. Από τις τότε στρατιωτικές επιχειρήσεις καθιερώθηκε ο όρος Ε.Ε. Ο πλούτος των εφαρμογών υπήρξε μεγάλος. Μόνο για το λεγόμενο Γραμμικό Προγραμματισμό (Γ.Π.) έχουν γραφτεί χιλιάδες βιβλία θεωρίας και εφαρμογών. Σημειωτέον ότι ο Γ.Π. έχει μία εύκολη απόδειξη λίγων γραμμών. Σύντομα μετά τον πόλεμο, ο κλάδος αξιοποιήθηκε από εταιρίες, βιομηχανίες, εργοστάσια, κατασκευές, κυβερνήσεις και επιχειρήσεις όλων των ειδών. Το πρόβλημα είναι γενικό και πρακτικό: Πώς αποφασίζω; Έχουν δοθεί βραβεία Nobel και έχουν δημιουργηθεί ολόκληρα τμήματα Ε.Ε. σε διάσημα Πανεπιστήμια και Πολυτεχνεία.

Στην παρούσα διπλωματική εργασία θα παρουσιάσουμε τις μεθόδους μη γραμμικού προγραμματισμού, αφού πρώτα κάνουμε μία εισαγωγή στην επιχειρησιακή έρευνα και τον γραμμικό προγραμματισμό.

Τέλος, αξίζει ένα μεγάλο ευχαριστώ στον επιβλέποντα καθηγητή της εργασίας αυτής, καθηγητή Γιάννη Κολέτσο, για την πίστη του, τις πολύτιμες συμβουλές και καθοδηγήσεις του, τις οποίες παρείχε απλόχερα και την αμέριστη στήριξη, που μου προσέφερε. Ευχαριστίες απευθύνω και στην οικογένειά μου, για την ενθάρρυνση, την ενίσχυση, με κάθε τρόπο, καθώς και για την υπομονή, καθ’ όλη τη διάρκεια της συγγραφής της διπλωματικής, αλλά και των χρόνων φοίτησής μου στην Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών. Είμαι, επίσης, ευγνώμων, σε όλους τους καθηγητές της σχολής, οι οποίοι μου μεταλαμπάδευσαν με προθυμία, υπευθυνότητα, χαρά και ζήλο τις γνώσεις τους.

Κεφάλαιο 1

Εισαγωγή

1.1Ιστορική αναδρομή

Από την εμφάνιση της βιομηχανικής επανάστασης (19^{ος} αιώνας) μέχρι σήμερα παρατηρείται μια αξιοσημείωτη ανάπτυξη στο μέγεθος και την πολυπλοκότητα των επιχειρήσεων και των οργανισμών. Οι μικρές επιχειρήσεις βαθμιαία εξελίχθηκαν σε εταιρίες συγκέντρωσης μεγάλων κεφαλαίων. Μέσα σε αυτά τα πλαίσια της επαναστατικής αλλαγής, αυξήθηκε εντυπωσιακά ο καταμερισμός των ευθυνών και η κατανομή της εργασίας μέσα στις επιχειρήσεις. Όμως, η αύξηση της εξειδίκευσης δημιούργησε καινούργια προβλήματα, τα οποία απασχολούν τις επιχειρήσεις και τους οργανισμούς μέχρι σήμερα. Για παράδειγμα, οι εξειδικευμένοι εργάτες, υπάλληλοι και επιστήμονες είχαν βαθιά γνώση του αντικειμένου με το οποίο ασχολούνταν, αλλά ελάχιστη γνώση της εργασίας των άλλων ειδικοτήτων και του τρόπου με τον οποίο οι δραστηριότητες και οι στόχοι τους συνδέονταν με τον κύριο οργανισμό. Επίσης, οι στόχοι και οι επιδιώξεις τους πολλές φορές έρχονταν σε αντίθεση με τους στόχους και τις επιδιώξεις των άλλων ειδικοτήτων. Τέλος, η κατανομή των διαθέσιμων πόρων στις διάφορες δραστηριότητες έγινε αρκετά δύσκολη. Δημιουργήθηκε, λοιπόν, η ανάγκη μιας επιστήμης η οποία όχι μόνο να έχει αντικείμενο την εύρεση του καλύτερου δυνατού τρόπου επίλυσης αυτών και άλλων παρόμοιων προβλημάτων, αλλά ταυτόχρονα να ικανοποιεί την προσπάθειά μας να προσεγγίσουμε επιστημονικά την διοίκηση των οργανισμών.

Παρουσιάστηκε, δηλαδή, η ανάγκη δημιουργίας της επιστήμης της **Επιχειρησιακής Έρευνας**. Το **1940** θεωρείται ως έτος γέννησης του νέου επιστημονικού κλάδου, όμως οι ρίζες της νέας αυτής επιστήμης (της Ε.Ε.) μπορούν να αναζητηθούν πολύ πριν. Το πρώτο βήμα στον τομέα έγινε από τον Charles Babbage (1791-1871). Χαρακτηρίστηκε ως ο ‘‘πατέρας της επιχειρησιακής έρευνας’’, επειδή η έρευνά του για το κόστος μεταφοράς και το κόστος για την ταξινόμηση της αλληλογραφίας οδήγησε στη δημιουργία του γενικού αγγλικού ‘‘Ταχυδρομείου της πένας’’ το 1840. Το 1917 ο Agner Krarup Erlang (1878-1929) μελετούσε προβλήματα σχετικά με το χρόνο απασχόλησης των τηλεφωνικών κέντρων και το 1920 ο Horace Levinson ασχολήθηκε με τη μελέτη προβλημάτων πωλήσεων και εμπορίου. Γιατί, όμως, αυτή η χρονολογία (1940) θεωρείται ως επίσημη αρχή του κλάδου; Τι συνέβη εκείνη την εποχή που προκάλεσε τη γέννηση της επιχειρησιακής έρευνας; Η έκρηξη του Β΄ Παγκοσμίου Πολέμου, το Σεπτέμβριο του 1939, συντάραξε την ανθρωπότητα. Για την τελική έκβαση του πολέμου επιστρατεύθηκαν όλες οι διαθέσιμες δυνάμεις της κοινωνίας, συμπεριλαμβανομένων και επιστημόνων, όπως μαθηματικών, φυσικών, στατιστικών, βιολόγων, ψυχιάτρων κ.ά. Έτσι λοιπόν, στο Ηνωμένο Βασίλειο επιστήμονες όπως οι Patrick Blackett (1897-1974), Cecil Gordon, Conrad Hal Waddington (1905-1975), Owen Wansbrough-Jones (1906-1983) και Frank Yates (1902-1994), αλλά και στις Ηνωμένες Πολιτείες ο George Dantzing (1914-2005), έψαχναν τρόπους, ώστε να λαμβάνουν καλύτερες αποφάσεις σε περιοχές όπως τα προγράμματα διοικητικής μέριμνας και τα προγράμματα εκπαίδευσης.

Ειδικότερα, οι διοικητές των ενόπλων δυνάμεων της Μεγάλης Βρετανίας χρειάζονταν βοήθεια για τη μελέτη της νέας τεχνολογίας των ραντάρ και την εξεύρεση των πιο αποτελεσματικών μεθόδων για τον εντοπισμό εχθρικών αεροσκαφών. Έτσι, το Υπουργείο British Air ίδρυσε το ερευνητικό κέντρο Bawdsey Manor Research Station στο Suffolk με διευθυντή του κέντρου για τα πρώτα χρόνια τον Robert Watson-Watt (1892-1973) (ως τότε υπεύθυνο του τμήματος ραδιοκυμάτων του Βρετανικού Εθνικού Εργαστηρίου) και στη συνέχεια τον Albert P. Rowe (1873-1948). Τα αποτελέσματα των τριών πρώτων επίσημων ασκήσεων με χρήση ραντάρ ήταν θετικά στην έγκαιρη προειδοποίηση, αλλά απογοητευτικά στις πληροφορίες εντοπισμού (tracking information) μετά το φιλτράρισμα και τη μετάδοση των στοιχείων από το δίκτυο. Στη συνέχεια, πραγματοποιήθηκε καινούργια άσκηση και προστέθηκαν τέσσερεις σταθμοί ραντάρ, όμως δημιουργήθηκε νέο πρόβλημα, αυτό του συντονισμού και συνδυασμού συχνά αντικρουόμενων πληροφοριών από τους διαφορετικούς σταθμούς. Ενώ τεχνικά τα συστήματα ραντάρ λειτουργούσαν αποτελεσματικά, υπήρχε πρόβλημα στο λειτουργικό (operational) κομμάτι, οπότε αποφασίστηκε έρευνα (research) προς αυτήν την κατεύθυνση. Εν όψει πολέμου, κάτι δραστικό έπρεπε να συμβεί. Έτσι, λοιπόν, το 1939 ιδρύεται το Stanmore Research Section

(SRS) και το 1941 το Operational Research Section της RAF (Royal Air Force), των οποίων οι αποφάσεις αποδείχτηκαν ιδιαίτερης σημασίας για την επιτυχημένη αναχαίτιση των Γερμανών στη μάχη της Βρετανίας.

Στην άλλη μεριά του Ατλαντικού ο P.M.S. Blackett, το 1942, συνέταξε ένα σχέδιο για τη δημιουργία του U.S. Navy Antisubmarine Warfare Operations Research Group (ASWORG). Οι εργασίες αυτής της ομάδας (που την αποκαλούσαν Blackett Circus) έφεραν σημαντικά αποτελέσματα με κυριότερο αυτό του προσδιορισμού του βέλτιστου μεγέθους των νηοπομπών (πλήθος φορτηγών πλοίων συνοδευόμενων από πολεμικά). Η αύξηση του μεγέθους των νηοπομπών καθόρισε τη νίκη στη μάχη του Ατλαντικού. Τον Οκτώβριο του 1942, η πρώτη ομάδα Επιχειρησιακών Ερευνητών από τις ΗΠΑ φτάνει στη Βρετανία για να συνεργαστεί με την 8η Μονάδα Βομβαρδιστικών. Στόχος τους ήταν η βελτιστοποίηση της ακρίβειας πλήξης στόχων. Η ομάδα βασισμένη σε μελέτη των προτύπων ρίψης βομβών και ύστερα από επιτόπου παρατήρηση, επρότεινε τα εξής:

i) Το πιο αποτελεσματικό πλήρωμα να ρίχνει πρώτο τις βόμβες του

ii) Όλες οι βόμβες να ρίχνονται ταυτόχρονα (in salvo)

iii) Τα αεροσκάφη να πετούν σε σχηματισμούς ακριβείας μειώνοντας με αυτόν τον τρόπο τη μεγάλη διασπορά των βομβών.

Η εφαρμογή των παραπάνω προτάσεων είχε σαν αποτέλεσμα τη βελτίωση της αποτελεσματικότητας πλήξης στόχων κατά 100%. Η νίκη της μάχης του Ατλαντικού και της μάχης της Βρετανίας ήταν μόνο δύο από τις σημαντικές εφαρμογές της επιχειρησιακής έρευνας κατά τη διάρκεια του Β΄ Παγκοσμίου πολέμου.

Μετά τον πόλεμο, η ανάγκη για την καλύτερη δυνατή αξιοποίηση των περιορισμένων διαθέσιμων πόρων ήταν φανερή. Παράλληλα, η ξαφνική και γρήγορη βιομηχανική ανάπτυξη δημιούργησε αρκετά προβλήματα λόγω της αυξανόμενης πολυπλοκότητας των οργανισμών και της εξειδίκευσης. Οι επιστήμονες και οι ερευνητές της επιχειρησιακής έρευνας που είχαν εργαστεί κατά τη διάρκεια του πολέμου δεν άργησαν να αντιληφθούν ότι τα προβλήματα αυτά είχαν την ίδια βάση με αυτά που αντιμετώπιζαν στο στρατό. Έτσι, η βιομηχανία παρακινούμενη από τη φανερή επιτυχία της επιχειρησιακής έρευνας στον στρατιωτικό τομέα άρχισε να ενδιαφέρεται για το νέο αυτό κλάδο. Στην Αμερική ιδρύθηκε το Operations Evaluation Group (OEG) σε συνεργασία με το MIT (Massachusetts Institute of Technology) και υπογράφηκε σύμβαση με την Douglas Aircraft Company για το έργο RAND (Research and Development), το οποίο στην ουσία επέκτεινε τη χρήση των ερευνητών επιχειρησιακής έρευνας για σημαντικό χρονικό διάστημα μετά τη λήξη του πολέμου. Μέχρι το 1951 η επιχειρησιακή έρευνα είχε επικρατήσει στη Μεγάλη Βρετανία και το 1957 ιδρύεται η Διεθνής Συνομοσπονδία Εταιρειών Επιχειρησιακών Ερευνών, που ονομάζεται διεθνώς ‘‘IFORS’’ (International Federation of Operations Research Societies). Δεδομένου ότι η Επιχειρησιακή Έρευνα γεννήθηκε κατά τον Β΄ Παγκόσμιο Πόλεμο και αναπτύχθηκε διεθνώς τη δεκαετία του 1960, η εμφάνισή της στη χώρα μας γίνεται πολύ σύντομα. Έτσι, ξεκινά τα πρώτα του βήματα και στη χώρα μας ένας νέος τρόπος σκέψης, πέρα από τους κλασσικούς και καθιερωμένους στο χώρο του Management. Το 1963, μία ομάδα πρωτοπόρων επιστημόνων ιδρύει την **‘‘Ελληνική Εταιρία Επιχειρησιακών Ερευνών’’** (Ε.Ε.Ε.Ε.), μία επιστημονική, μη κερδοσκοπική εταιρία, που έχει ως κύριο σκοπό να προαγάγει και να διαδώσει την Επιχειρησιακή Έρευνα στην Ελλάδα.

Ένας από τους παράγοντες που έπαιξαν σημαντικό ρόλο στη ραγδαία ανάπτυξη της επιχειρησιακής έρευνας, είναι η βελτίωση των τεχνικών της επιχειρησιακής έρευνας. Για παράδειγμα, η μέθοδος Simplex αναπτύχθηκε από τον G. Dantzing το 1947 και βοήθησε στην επίλυση προβλημάτων γραμμικού προγραμματισμού. Επίσης, πολλά από τα εργαλεία της επιχειρησιακής έρευνας, όπως ο γραμμικός προγραμματισμός, ο δυναμικός προγραμματισμός και η θεωρία ουρών αναμονής αναπτύχθηκαν πριν το τέλος της δεκαετίας του 1950. Παράλληλα, η πρόοδος της τεχνολογίας των ηλεκτρονικών υπολογιστών έκανε δυνατή την παροχή πληθώρας ενημερωμένων πληροφοριών στα στελέχη της διοίκησης, τα οποία απαιτούσαν επεξεργασία. Η ανάγκη, λοιπόν, για ταχεία και αποτελεσματική αντιμετώπιση των περίπλοκων επιχειρησιακών προβλημάτων και η πρόοδος της τεχνολογίας οδήγησαν σε μία σταδιακή εμφάνιση και αξιοποίηση της Επιχειρησιακής Έρευνας στο χώρο της βιομηχανίας, καθώς και στους χώρους παροχής υπηρεσιών, προγραμματισμού, κ.λπ.

1.2Τι είναι η Επιχειρησιακή Έρευνα

Η **Επιχειρησιακή Έρευνα** είναι, όπως είδαμε, ένα κύριο εργαλείο, το οποίο χρησιμοποιείται για την άσκηση της διοίκησης και ειδικότερα για τη λήψη αποφάσεων με τη βοήθεια των ηλεκτρονικών υπολογιστών. Το όνομά της προέρχεται από τη μετάφραση του Αγγλικού όρου ‘‘Operational Research’’ ή όπως συνηθίζεται στην Αμερική ‘‘Operations Research’’ και σημαίνει ‘‘έρευνα στις επιχειρήσεις’’. Η απόδοση αυτή του όρου στα Ελληνικά μάς λέει κάτι τόσο για την προσέγγιση όσο και για το πεδίο εφαρμογής του τομέα, όμως θα μπορούσαμε να πούμε πως δεν είναι και η ακριβέστερη δυνατή. Η απόδοση ‘‘Λειτουργική Έρευνα’’ θα ήταν πιο επιτυχημένη, αφού αποδίδει πληρέστερα το περιεχόμενο του θέματος, δηλαδή έρευνα πάνω στις λειτουργίες ή στη λειτουργία πολύπλοκων συστημάτων αποτελούμενων από ανθρώπους και μηχανές. Στο στρατό, όμως, απ’ όπου και ξεκίνησε η ανάπτυξη και η εφαρμογή της επιχειρησιακής έρευνας, οι κυριότερες λειτουργίες ονομάζονται επιχειρήσεις. Επομένως, ίσως για αυτό το λόγο καθιερώθηκε αυτός ο όρος γενικότερα στην Ελλάδα. Επίσης, ο όρος ‘‘Επιχειρησιακή Έρευνα’’ είναι πιο κατανοητός από τους ανθρώπους της πράξεως στον επιχειρησιακό τομέα.

Τα τελευταία χρόνια χρησιμοποιούνται διεθνώς και οι εναλλακτικές ονομασίες **Διοικητική Επιστήμη** (Management Science), **Επιστήμη Αποφάσεων** (Decision Science) και Ανάλυση Συστημάτων (Systems Analysis). Επίσης, χρησιμοποιείται συχνά η συνδυαστική ονομασία **Operations Research & Management Science** (OR/MS) για να υποδηλώσει ότι οι δύο όροι σημαίνουν περίπου το ίδιο.

Τι σημαίνει, όμως, ο όρος Επιχειρησιακή Έρευνα; Ο καλύτερος τρόπος να απαντήσουμε σ’ αυτήν την ερώτηση θα ήταν να δώσουμε έναν ορισμό. Κατά καιρούς, έχουν προταθεί διάφοροι ορισμοί, οι οποίοι δεν διαφέρουν ιδιαίτερα μεταξύ τους. Αναφέρουμε μερικούς από τους σημαντικότερους.

Ο R. Watson-Watt, ο οποίος μαζί με τον A. P. Rowe, φαίνεται ότι πρότεινε το όνομα ‘‘Operational Research’’ και ο οποίος ήταν ένας από τους πρωτεργάτες της εισαγωγής και ανάπτυξης του θέματος στην Βρετανική αεροπορία έδωσε τον εξής ορισμό:

‘‘Η Επιχειρησιακή Έρευνα αποσκοπεί στο να ερευνήσει ποσοτικά εάν ένας οργανισμός παίρνει από τη λειτουργία τού εξοπλισμού του τη βέλτιστη δυνατή συνεισφορά, σε σχέση με τον ολικό αντικειμενικό σκοπό του, ποιες αλλαγές σε εξοπλισμό και μεθόδους απαιτούνται για τη βελτίωση των αποτελεσμάτων με το μικρότερο δυνατό κόστος σε προσπάθεια και χρόνο και τέλος σε ποιο βαθμό μεταβολές στους επιμέρους αντικειμενικούς σκοπούς θα συνεισέφεραν στην πιο οικονομική και έγκαιρη εκτέλεση του ολικού στρατηγικού αντικειμενικού σκοπού’’.

Από την Εταιρεία Επιχειρησιακής Έρευνας της Μεγάλης Βρετανίας (Operational Research Society) έχει προταθεί ο παρακάτω ορισμός:

‘‘Επιχειρησιακή Έρευνα είναι η εφαρμογή της σύγχρονης επιστήμης πάνω σε πολύπλοκα προβλήματα, τα οποία ανακύπτουν στη διεύθυνση και τη διοίκηση μεγάλων συστημάτων, αποτελούμενων από ανθρώπους, μηχανές, υλικά και κεφάλαια, στη Βιομηχανία, τις Επιχειρήσεις, τις Κυβερνητικές Υπηρεσίες και την Άμυνα.

Η χαρακτηριστική της μεθοδολογία συνίσταται στην ανάπτυξη επιστημονικού μοντέλου τού υπό μελέτη συστήματος, που περιλαμβάνει μετρήσεις τυχαίων παραγόντων και με το οποίο προβλέπει και συγκρίνει τα αποτελέσματα εναλλακτικών αποφάσεων, στρατηγικών και ελέγχων.

Ο σκοπός της είναι να βοηθήσει τη διοίκηση να καθορίσει την πολιτική και τις ενέργειές της επιστημονικά (κατά τον καλύτερο δυνατό τρόπο)’’.

Ο ορισμός που έχει προταθεί από τους Russell Lincoln Ackoff (1919-2009) και Maurice W. Sasieni επισημαίνει ιδιαίτερα ορισμένα χαρακτηριστικά της επιστημονικής φύσεως της Επιχειρησιακής Έρευνας:

‘‘Επιχειρησιακή Έρευνα μπορεί να θεωρηθεί ότι είναι: η εφαρμογή επιστημονικών μεθόδων από μικτές ομάδες σε προβλήματα που αφορούν τον έλεγχο οργανωμένων συστημάτων (αποτελούμενων από ανθρώπους και μηχανές) κατά τρόπο, ώστε να παρέχουν λύσεις που εξυπηρετούν κατά τον καλύτερο δυνατό τρόπο τους σκοπούς του οργανισμού ως συνόλου’’.

Ο ιδιαίτερα εύστοχος ορισμός της Ε.Ε.Ε.Ε. είναι:

‘‘Επιχειρησιακή Έρευνα είναι η επιστημονική προετοιμασία των αποφάσεων της Διοικήσεως (με την επιστημονική ανάλυση των δεδομένων και τη δημιουργία μαθηματικών προτύπων)’’.

1.3 Βασικά στάδια της Επιχειρησιακής Έρευνας

Η επιστημονική μεθοδολογία που ακολουθείται στην Επιχειρησιακή Έρευνα είναι σχεδόν πάντα η ίδια, ανεξάρτητα από το πεδίο εφαρμογής ή το μοντέλο που θα χρησιμοποιηθεί. Τα στάδια είναι τα εξής:

Ανάλυση του Συστήματος

Στο στάδιο αυτό, ο στόχος είναι να κατανοήσουμε το σύστημα που θα μελετήσουμε. Για το λόγο αυτό, προσδιορίζουμε τη δομή του και τον τρόπο λειτουργίας του. Στη συνέχεια, το αναλύουμε στα υποσυστήματά του, εντοπίζουμε τα σημεία στα οποία εμείς μπορούμε να επηρεάσουμε τη λειτουργία του και προσδιορίζουμε τρόπους (στρατηγικές), τους οποίους μπορούμε να εφαρμόσουμε προκειμένου να επηρεάσουμε το υπό μελέτη σύστημα. Με τον τρόπο αυτό, αποκτούμε σαφή αντίληψη του προβλήματος που έχουμε να αντιμετωπίσουμε και αναγνωρίζουμε τις μεταβλητές – παραμέτρους του συστήματος, καθώς και τους διάφορους περιορισμούς που επιβάλλονται από τη δομή του, τη λειτουργία του, ή το περιβάλλον.

Διατύπωση Στόχων

Στο στάδιο αυτό, θέτουμε τους στόχους που θέλουμε να πετύχουμε, π.χ. μεγιστοποίηση του κέρδους, ελαχιστοποίηση του κόστους, βελτίωση της παραγωγικότητας. Η φάση αυτή είναι ιδιαίτερα σημαντική, καθώς από τη διατύπωση των σωστών στόχων εξαρτάται η επιτυχία και η εφαρμογή των λύσεων που θα προταθούν. Η διατύπωση των στόχων δεν είναι πάντα εύκολη, αφού αρκετές φορές υπάρχουν πολλαπλοί στόχοι και πρέπει να τους ιεραρχήσουμε ή κατά κάποιο τρόπο να τους συμπτύξουμε. Για παράδειγμα, σε μία εταιρεία που παράγει και πουλάει προϊόντα, το τμήμα πωλήσεων επιδιώκει να βρει ποια προϊόντα θα αυξήσουν τις πωλήσεις, το τμήμα παραγωγής ενδιαφέρεται για την καλύτερη αξιοποίηση των πόρων, τη μεγιστοποίηση της παραγωγικότητας και την καλύτερη απασχόληση του εργατικού δυναμικού. Από την άλλη, οι μέτοχοι της εταιρείας θέλουν τη μεγιστοποίηση των κερδών και του μερίσματος. Προφανώς, όλοι αυτοί οι στόχοι δεν επιτυγχάνονται με την ίδια στρατηγική και θα πρέπει να υπάρξει κάποια σειρά προτεραιότητας.

Διατύπωση του Μοντέλου

Στο στάδιο αυτό δημιουργούμε μία απλουστευμένη αναπαράσταση του πραγματικού συστήματος, με σκοπό να τη μελετήσουμε, να αναλύσουμε την επίδραση διαφόρων παραγόντων στους στόχους που έχουν τεθεί και να επιλέξουμε την καλύτερη στρατηγική. Η κατασκευή του μαθηματικού μοντέλου έχει ως αποτέλεσμα τη μετατροπή του ορισμού του προβλήματος σε μαθηματικές σχέσεις και αποτελεί μία κατά προσέγγιση αναπαράσταση του προβλήματος. Το μοντέλο αυτό είναι συνήθως ένα σύνολο από ποσοτικές σχέσεις ή εντολές στον υπολογιστή, που εκφράζουν τους στόχους του προβλήματος και τους περιορισμούς του περιβάλλοντος. Η διαδικασία της διατύπωσης του μοντέλου χωρίζεται σε τρεις φάσεις:

- ✓ Διατύπωση ορισμένων υποθέσεων, οι οποίες απλουστεύουν το πρόβλημα και οι οποίες είναι λίγο έως πολύ ρεαλιστικές. Αυτό γίνεται για να κάνει πιο εφικτή και εύκολη την ανάλυση και την επίλυση του προβλήματος.

- ✓ Διατύπωση εκείνων των μαθηματικών σχέσεων ή εντολών στον υπολογιστή, οι οποίες εκφράζουν τις σχέσεις μεταξύ των συντελεστών του συστήματος, των στόχων, των μεταβλητών και του περιβάλλοντος.

- ✓ Επιβεβαίωση του μοντέλου με δοκιμαστική χρήση του σε ένα ‘‘απλό’’ πρόβλημα. Αυτό γίνεται για να ελέγξουμε την ακρίβεια των υποθέσεων (φάση 1) και των σχέσεων/εντολών (φάση 2) που διατυπώθηκαν παραπάνω. Σε περίπτωση που τα αποτελέσματα δεν είναι ικανοποιητικά, οι φάσεις (1 και 2) επαναλαμβάνονται.

Η διαμόρφωση του μοντέλου είναι ένα δύσκολο έργο που αν αποτύχει, οδηγεί στη λήψη λανθασμένων αποφάσεων. Ο αναλυτής ενός συστήματος διατρέχει τον κίνδυνο να εντοπίσει τη σωστή λύση σε άλλο πρόβλημα (το προτεινόμενο μοντέλο δεν αποτελεί σωστή αναπαράσταση του προβλήματος), ενώ η μέθοδος βελτιστοποίησης της αντικειμενικής συνάρτησης είναι σωστή. Στην αντίθετη περίπτωση, κινδυνεύει να δώσει λάθος λύση σε σωστό πρόβλημα. Εκτός των άλλων, ο αναλυτής πρέπει να φροντίσει για την ελάττωση των διαστάσεων του προβλήματος, δηλαδή τον περιορισμό των μεταβλητών σε εκείνες που είναι πραγματικά σημαντικές για το πρόβλημα, δεδομένου ότι από αυτό εξαρτάται το κόστος επίλυσής του.

Επίλυση του Μοντέλου

Η επίλυση του μαθηματικού μοντέλου χρησιμοποιεί διάφορες τεχνικές για τον εντοπισμό της βέλτιστης λύσης του προβλήματος. Με τις λέξεις «λύση του προβλήματος», εννοούμε τον προσδιορισμό της στρατηγικής που θα ακολουθήσουμε. Οι διάφορες μέθοδοι που χρησιμοποιούνται εδώ στηρίζονται σε Ανώτερα Μαθηματικά (διαφορικό και ολοκληρωτικό λογισμό, αριθμητική ανάλυση, γραμμική άλγεβρα, κλασικές μεθόδους βελτιστοποίησης, λογισμό των μεταβολών), στη θεωρία Πιθανοτήτων (κατανομές πιθανοτήτων, ανελίξεις Markov) και τη Στατιστική (περιγραφική στατιστική, στατιστική συμπερασματολογία, εκτίμηση παραμέτρων, ανάλυση παλινδρόμησης, αλληλοσυσχέτιση, ανάλυση μεταβλητότητας, παραγοντική ανάλυση, χρονοσειρές) ή σε Μεθόδους και Θεωρίες της επιχειρησιακής έρευνας. Οι τελευταίες είναι οι πιο εύκολες και συχνά χρησιμοποιούμενες, αφού είναι κατά κανόνα αριθμητικές, επαναληπτικές μέθοδοι και βασίζονται στη χρήση αλγορίθμων που υλοποιούνται με υπολογιστή.

Η πιο χαρακτηριστική τεχνική της επιχειρησιακής έρευνας είναι ο γραμμικός προγραμματισμός (linear programming). Σχεδιάστηκε για μοντέλα με αυστηρά γραμμικούς περιορισμούς. Άλλες τεχνικές είναι ο ακέραιος προγραμματισμός (integer programming), στον οποίο οι μεταβλητές θεωρούνται ακέραιες, ο δυναμικός προγραμματισμός (dynamic programming), στον οποίο το πρόβλημα μπορεί να διαιρεθεί σε μικρότερα υπό-προβλήματα και ο μη γραμμικός προγραμματισμός (non-linear programming), στον οποίο οι συναρτήσεις των μοντέλων είναι μη γραμμικές και με τον οποίο θα ασχοληθούμε στην παρούσα διπλωματική εργασία. Επίσης, ο προγραμματισμός δικτύων (network programming), τα δέντρα αποφάσεων (decision trees), η διαχείριση αποθεμάτων (inventory control), η θεωρία ουρών αναμονής (queuing theory) και η θεωρία παιγνίων (game theory) είναι αρκετά χρήσιμες μέθοδοι της επιχειρησιακής έρευνας. Πρέπει να τονίσουμε ότι οι παραπάνω τεχνικές δεν είναι παρά μόνο μια μικρή λίστα από τα διαθέσιμα εργαλεία της επιχειρησιακής έρευνας.

Η επιλογή της κατάλληλης μεθόδου ή θεωρίας της επιχειρησιακής έρευνας εξαρτάται από τον τύπο και την πολυπλοκότητα του μαθηματικού μοντέλου. Αν το μοντέλο ταιριάζει σε κάποιο από τα γνωστά μαθηματικά μοντέλα, όπως π.χ. αυτό του γραμμικού προγραμματισμού, τις περισσότερες φορές μπορούμε να βρούμε μία εφικτή λύση, με τη χρήση αλγορίθμων. Εναλλακτικά, αν οι μαθηματικές σχέσεις είναι τόσο περίπλοκες, ώστε να μην επιτρέπουν την εύρεση μιας αναλυτικής λύσης, οι ομάδες της επιχειρησιακής έρευνας αναγκάζονται να απλοποιήσουν το μοντέλο και να χρησιμοποιήσουν ευρετικές τεχνικές (heuristics) ή μεθόδους προσομοίωσης (simulation). Σε ορισμένες περιπτώσεις, μάλιστα, ίσως χρειαστεί συνδυασμός των προαναφερθεισών μεθόδων.

Ανάλυση Ευαισθησίας (Sensitivity Analysis)

Η λύση, την οποία μας υπέδειξε το μοντέλο στο προηγούμενο στάδιο, ισχύει για τις παραμέτρους του περιβάλλοντος (τιμές, δυναμικότητα κ.λπ.) που ορίσαμε αρχικά, όταν διατυπώναμε το μοντέλο. Ωστόσο, πριν υλοποιήσουμε τη στρατηγική που υποδεικνύει το μοντέλο, ενδιαφερόμαστε συχνά να γνωρίζουμε ποια επίπτωση θα έχει στην άριστη στρατηγική μία απρόβλεπτη αλλαγή στο περιβάλλον. Αυτή η ανάλυση της λύσης ονομάζεται στη βιβλιογραφία **ανάλυση ευαισθησίας**.

Είναι μια μέθοδος που προσδιορίζει την ευαισθησία της λύσης, όταν το μοντέλο υποβάλλεται σε μεταβολές των παραμέτρων του.

Η ανάλυση ευαισθησίας χρειάζεται ιδιαίτερη προσοχή, όταν οι παράμετροι του μοντέλου δεν μπορούν να υπολογιστούν με ακρίβεια. Σε αυτές τις περιπτώσεις, είναι σημαντικό να μελετήσουμε τη συμπεριφορά της λύσης σε μια ‘‘γειτονιά’’ των παραμέτρων του μοντέλου. Το στάδιο αυτό είναι σημαντικό, διότι παρέχει χρήσιμη πληροφόρηση και μπορεί να επηρεάσει ουσιαστικά στην επιλογή της στρατηγικής που θα ακολουθήσουμε.

Υλοποίηση της Λύσης

Έχοντας επιλέξει τη στρατηγική που θα ακολουθήσουμε, πρέπει τώρα να τη θέσουμε σε εφαρμογή. Το στάδιο αυτό είναι συχνά το δυσκολότερο. Η υλοποίηση και διατήρηση της λύσης ενός μοντέλου, περιλαμβάνει τη μετατροπή των αποτελεσμάτων σε λειτουργικές οδηγίες, παρουσιασμένες με κατανοητό τρόπο στα άτομα που θα διαχειριστούν το προτεινόμενο σύστημα, έτσι ώστε η βελτίωση που επιτεύχθηκε να υλοποιηθεί στο πραγματικό σύστημα και να διατηρηθεί στο μέλλον. Το βάρος αυτού του σταδίου επωμίζεται κυρίως η ομάδα της επιχειρησιακής έρευνας, αφού είναι πιθανό να προκύψουν προβλήματα τα οποία δεν είχαν προβλεφθεί κατά τη διάρκεια της έρευνας.

1.4Εφαρμογές της Επιχειρησιακής Έρευνας

Συνοπτικά, θα μπορούσαμε να πούμε πως η επιχειρησιακή έρευνα ασχολείται με την άριστη λήψη αποφάσεων σε προσδιοριστικά και πιθανολογικά συστήματα, που προκύπτουν μέσα από πραγματικά προβλήματα. Διακρίνοντας τα προβλήματα αυτά ως προς το περιεχόμενό τους, παραθέτουμε μερικά παραδείγματα.

Βιομηχανία-Παραγωγή

Χωροθέτηση εργοστασίου, προγραμματισμός προμηθειών ή παραγωγής, έλεγχος αποθεμάτων πρώτων υλών ή προϊόντων, έλεγχος ποιότητας, ανανέωση μηχανολογικού εξοπλισμού, ανάλυση αξιοπιστίας, εξισορρόπηση γραμμής παραγωγής.

Εμπόριο

Καθορισμός βέλτιστης σύνθεσης παραγωγής, βέλτιστη στρατηγική διαφημίσεως και τιμολογήσεως προϊόντων, προγραμματισμός πωλήσεων, προγραμματισμός μεταφοράς και διανομής προϊόντων, προσδιορισμός θέσεως και αριθμού αποθηκών, σύνθεση μεταφορικού στόλου.

Οικονομικά

Χρηματοοικονομικός προγραμματισμός, καθορισμός πιστωτικής πολιτικής, προϋπολογισμός, βελτιστοποίηση χρηματορροών.

Προσωπικό

Ανάλυση & αξιολόγηση προσωπικού, προγραμματισμός προσωπικού, ανάλυση αιτιών απουσίας, πρόληψη ατυχημάτων.

Προβλήματα σαν κι αυτά όμως, απασχολούν τους περισσότερους οργανισμούς, μεταξύ των οποίων τις εμπορικές και βιομηχανικές επιχειρήσεις. Συνεπώς, τόσο οι μεγάλες εταιρείες, όσο και πολλοί από τους μικρούς βιομηχανικούς οργανισμούς διαθέτουν στο μόνιμο προσωπικό τους ομάδες επιχειρησιακής έρευνας. Πολλές βιομηχανίες, όπως βιομηχανίες κατασκευής αεροσκαφών, αυτοκινήτων, μέσων επικοινωνίας, ηλεκτρονικών υπολογιστών, τροφίμων, μεταλλουργίας, μεταλλευμάτων, χαρτιού, πετρελαίου και μέσων μεταφοράς, χρησιμοποίησαν και χρησιμοποιούν ευρέως την επιχειρησιακή έρευνα. Επίσης, οι κυβερνητικές υπηρεσίες, χρηματοπιστωτικοί οργανισμοί και νοσοκομεία τη χρησιμοποιούν όλο και περισσότερο.

Κεφάλαιο 2: Μη γραμμικός Προγραμματισμός

Ο θεμελιώδης ρόλος του γραμμικού προγραμματισμού στην Ε.Ε. είναι δεδομένος και αναμφισβήτητος. Μια βασική παραδοχή τού γραμμικού προγραμματισμού είναι ότι όλες οι συναρτήσεις του (αντικειμενική συνάρτηση και συναρτήσεις περιορισμών), είναι γραμμικές. Αυτή η υπόθεση έχει πολλά ουσιαστικά πρακτικά προβλήματα, στις περισσότερες περιπτώσεις. Στην πραγματικότητα, πολλοί οικονομολόγοι έχουν διαπιστώσει ότι ένας βαθμός μη γραμμικότητας είναι ο κανόνας και όχι η εξαίρεση σε οικονομικά προβλήματα σχεδιασμού. Ως εκ τούτου, είναι απαραίτητο να ασχοληθούμε άμεσα με τα προβλήματα μη γραμμικού προγραμματισμού, άρα να στρέψουμε την προσοχή μας σε αυτόν τον σημαντικό τομέα.

Στη γενικευμένη μορφή, σε ένα πρόβλημα μη γραμμικού προγραμματισμού ψάχνουμε ένα $x = (x_1, x_2, \ldots, x_n)$ τέτοιο, ώστε:

Να μεγιστοποιήσουμε την $f(x)$ [Max $f(x)$],

υποκείμενη στους περιορισμούς:

$g(x_i) \leq b_i$, για $i = 1, \ldots, m$

και $x \geq 0,$

όπου οι $f(x)$ και $g(x)$ είναι δοσμένες μη γραμμικές συναρτήσεις των n μεταβλητών απόφασης.

Δεν υπάρχει αλγόριθμος, που να λύνει κάθε συγκεκριμένο πρόβλημα αυτής της μορφής. Ωστόσο, σημαντική πρόοδος έχει γίνει για ορισμένες σημαντικές ειδικές περιπτώσεις του προβλήματος, κάνοντας διάφορες υποθέσεις σχετικά με αυτές τις συναρτήσεις, και η έρευνα συνεχίζεται πολύ ενεργά. Αυτό το πεδίο είναι μεγάλο και δεν έχουμε τον χώρο για να το ερευνήσουμε πλήρως. Ωστόσο, παρουσιάζουμε μερικές εφαρμογές και -στη συνέχεια- εισάγουμε ορισμένες από τις βασικές ιδέες για την επίλυση ορισμένων σημαντικών τύπων προβλημάτων μη γραμμικού προγραμματισμού.

Τα ακόλουθα παραδείγματα απεικονίζουν μερικά από τα πολλά σημαντικά είδη των προβλημάτων, στα οποία εφαρμόζεται ο μη γραμμικός προγραμματισμός.

2.1 Το πρόβλημα σύνθεσης προϊόντων, με ελαστικότητα τιμής

Σε προβλήματα σύνθεσης του προϊόντος, ο στόχος είναι να καθοριστεί ο βέλτιστος συνδυασμός των επιπέδων παραγωγής για τα προϊόντα τής επιχείρησης, δεδομένων των περιορισμών σχετικά με τους πόρους που απαιτούνται για την παραγωγή των εν λόγω προϊόντων, ώστε να μεγιστοποιηθεί το συνολικό κέρδος τής επιχείρησης. Σε ορισμένες περιπτώσεις, υπάρχει μια σταθερή μονάδα κέρδους που σχετίζεται με κάθε προϊόν, οπότε η τελική αντικειμενική συνάρτηση είναι γραμμική. Ωστόσο, σε πολλά προβλήματα σύνθεσης του προϊόντος, ορισμένοι παράγοντες εισάγουν μη γραμμικότητες στην αντικειμενική συνάρτηση.

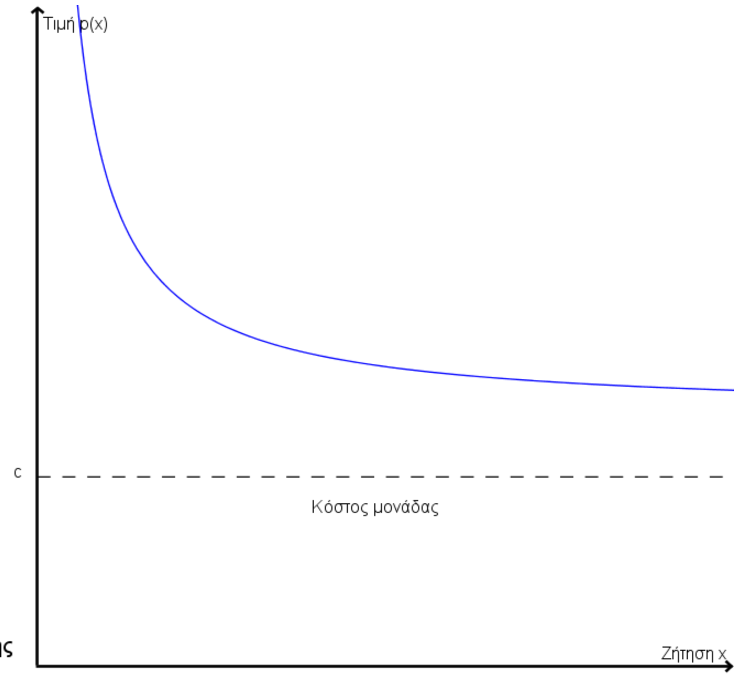
Για παράδειγμα, ένας μεγάλος κατασκευαστής μπορεί να συναντήσει ελαστικότητα των τιμών, όπου η ποσότητα του προϊόντος που μπορεί να πωληθεί έχει μία αντιστρόφως ανάλογη σχέση με την τιμή που χρεώνεται. Έτσι, η καμπύλη τιμών-ζήτησης για ένα τυπικό προϊόν θα μπορούσε να μοιάζει με αυτήν που φαίνεται στο Σχ.2.1.1, όπου $p(x)$ είναι η τιμή που απαιτείται προκειμένου να είναι σε θέση να πωλήσει x μονάδες. Τα κέρδη τής επιχείρησης από την παραγωγή και την πώληση x μονάδων του προϊόντος, τότε θα ήταν τα έσοδα των πωλήσεων, $xp(x)$, μείον το κόστος παραγωγής και διανομής. Ως εκ τούτου, εάν η μονάδα κόστους για την παραγωγή και τη διανομή του προϊόντος καθορίζεται σε c (βλέπε τη διακεκομμένη γραμμή στο Σχ.2.1.1), το κέρδος της επιχείρησης από την παραγωγή και την πώληση x μονάδων δίνεται από τη μη γραμμική συνάρτηση:

$$P(x) = xp(x) - cx,$$

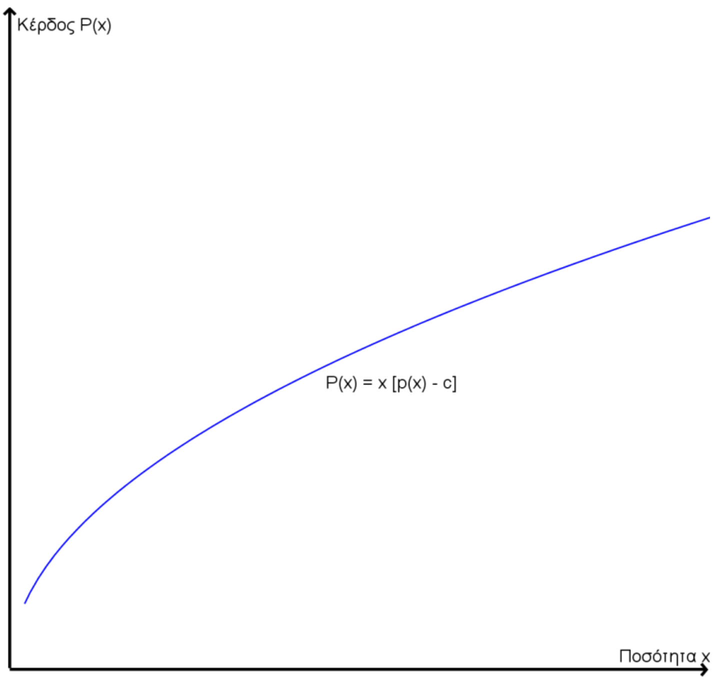
όπως έχει σχεδιαστεί στο Σχ.2.1.2. Αν κάθε ένα από τα n προϊόντα της εταιρείας έχει παρόμοια συνάρτηση κέρδους, έστω $P_j(x_j)$ για την παραγωγή και πώληση x_j μονάδων προϊόντος j $(j = 1,2,\ldots,n)$ τότε η ολική αντικειμενική συνάρτηση είναι:

$$f(x) = \sum_{j=1}^{n} P_j\left(x_j\right),$$

άθροισμα μη γραμμικών συναρτήσεων.



Σχήμα 2.1.1
Καμπύλη Τιμής-Ζήτησης



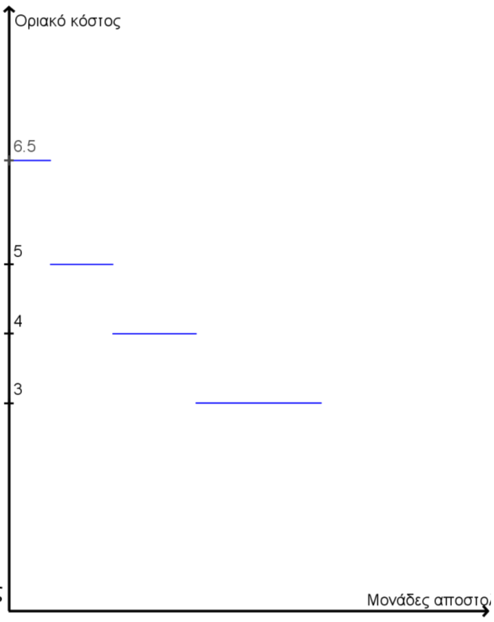
Σχήμα 2.1.2
Συνάρτηση Κέρδους

Ένας άλλος λόγος από τον οποίο μπορεί να προκύψουν μη γραμμικότητες στην αντικειμενική συνάρτηση, είναι το γεγονός ότι το οριακό κόστος της παραγωγής μιας άλλης μονάδας ενός συγκεκριμένου προϊόντος, ποικίλλει ανάλογα με το επίπεδο της παραγωγής. Για παράδειγμα, το οριακό κόστος μπορεί να μειωθεί, όταν το επίπεδο της παραγωγής αυξάνεται, λόγω της επίδρασης της εκμάθησης πάνω στην καμπύλη παραγωγής (πιο αποδοτική παραγωγή όταν υπάρχει μεγαλύτερη εμπειρία). Από την άλλη πλευρά, μπορεί αντιθέτως να το αυξήσει, διότι ενδέχεται να χρειαστούν ειδικά μέτρα, όπως υπερωρίες ή πιο δαπανηρές εγκαταστάσεις παραγωγής για την περαιτέρω αύξηση της παραγωγής.

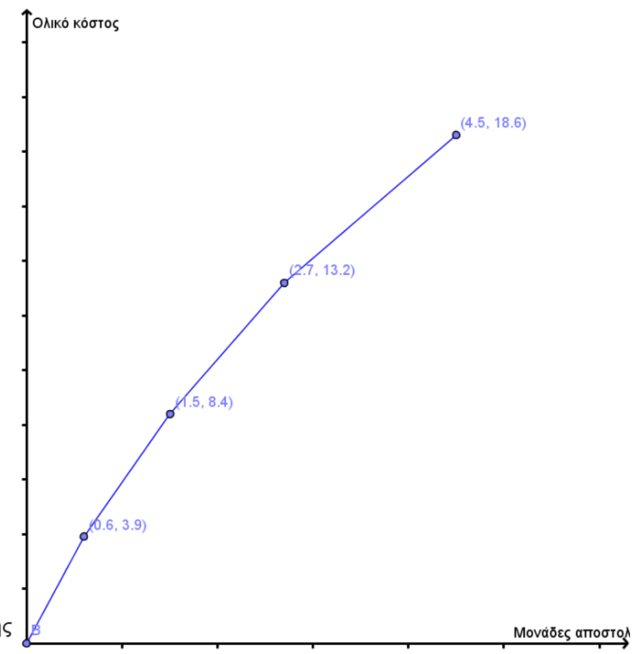
Μη γραμμικότητες, επίσης, μπορεί να προκύψουν στις $g_i(x)$ (συναρτήσεις περιορισμών) με έναν παρόμοιο τρόπο. Για παράδειγμα, εάν υπάρχει ένας περιορισμός τού προϋπολογισμού για το συνολικό κόστος παραγωγής, η συνάρτηση κόστους θα είναι μη γραμμική εάν το οριακό κόστος παραγωγής ποικίλλει, όπως ακριβώς περιγράφεται. Για τους περιορισμούς για τα άλλα είδη των πόρων, η $g_i(x)$ θα είναι μη γραμμική, όταν η χρήση της αντίστοιχης πηγής δεν είναι απολύτως ανάλογη προς τα επίπεδα παραγωγής των αντίστοιχων προϊόντων.

2.2 Το πρόβλημα μεταφοράς, με εκπτώσεις όγκου στα έξοδα αποστολής

Μια τυπική εφαρμογή τού προβλήματος της μεταφοράς είναι να καθορίσει ένα βέλτιστο σχέδιο, για την αποστολή εμπορευμάτων από διάφορες πηγές σε διάφορους προορισμούς, δεδομένων των περιορισμών της προσφοράς και της ζήτησης, προκειμένου να ελαχιστοποιηθεί το συνολικό κόστος της αποστολής. Θεωρούμε ότι το κόστος ανά μονάδα που αποστέλλεται από μια δεδομένη πηγή σε ένα δεδομένο προορισμό είναι σταθερό, ανεξάρτητα από την ποσότητα που αποστέλλεται. Στην πραγματικότητα, το κόστος αυτό δεν μπορεί να καθοριστεί. Μερικές φορές, είναι διαθέσιμες εκπτώσεις όγκου για τις μεγάλες αποστολές, έτσι ώστε το οριακό κόστος της μεταφοράς μίας μονάδας να μπορούσε να ακολουθήσει ένα μοτίβο, όπως αυτό που φαίνεται στο Σχ.2.2.1. Το προκύπτον κόστος αποστολής x μονάδων τότε δίνεται από μία μη γραμμική συνάρτηση $C(x)$, η οποία είναι μια τμηματικά γραμμική συνάρτηση με κλίση ίση με το οριακό κόστος, όπως αυτό φαίνεται στο Σχ.2.2.2. [Η συνάρτηση στο Σχ.2.2.2 αποτελείται από ένα ευθύγραμμο τμήμα με κλίση 6.5 από (0, 0) έως (0.6, 3.9), ένα δεύτερο ευθύγραμμο τμήμα με κλίση από 5 από (0.6, 3.9) έως (1.5, 8.4), ένα τρίτο ευθύγραμμο τμήμα με κλίση 4 από (1.5, 8.4) έως (2.7, 13.2), και ένα τέταρτο ευθύγραμμο τμήμα με κλίση 3 από (2.7, 13.2) έως (4.5, 18.6).] Ως εκ τούτου, αν κάθε συνδυασμός της πηγής και του προορισμού έχει παρόμοια συνάρτηση εξόδων αποστολής, έτσι ώστε το κόστος



Σχήμα 2.2.1
Οριακό κόστος αποστολής



Σχήμα 2.2.2
Συνάρτηση κόστους αποστολής

μεταφοράς x_{ij} μονάδων από την πηγή i $(i = 1, 2,..., m)$ στον προορισμό j $(j = 1, 2,..., n)$ δίνεται από μια μη γραμμική συνάρτηση $C_{ij}(x_{ij})$, τότε η συνολική αντικειμενική συνάρτηση προς ελαχιστοποίηση είναι:

$$f(x) = \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij}(x_{ij}).$$

2.3 Επιλογή χαρτοφυλακίου, με ριψοκίνδυνους τίτλους (χρεόγραφα)

Είναι κοινή πρακτική για τους επαγγελματίες διαχειριστές μεγάλων χαρτοφυλακίων μετοχών να χρησιμοποιούν μοντέλα υπολογιστών, που βασίζονται εν μέρει σε μη γραμμικό προγραμματισμό για να τους καθοδηγούν. Επειδή, οι επενδυτές ανησυχούν τόσο για την *αναμενόμενη απόδοση* (κέρδος) και τον *κίνδυνο* (ρίσκο), που συνδέεται με τις επενδύσεις τους, χρησιμοποιούν μη γραμμικό προγραμματισμό για τον προσδιορισμό ενός χαρτοφυλακίου που, υπό ορισμένες προϋποθέσεις, παρέχει μια βέλτιστη συμφωνία (συνάρτηση) μεταξύ των δύο αυτών παραγόντων. Αυτή η προσέγγιση βασίζεται σε μεγάλο βαθμό στο σπάσιμο της έρευνας σε βήματα, που έγινε από τους Harry Markowitz και William Sharpe, κάτι που τους βοήθησε να κερδίσουν το βραβείο Νόμπελ, το 1990, στα Οικονομικά.

Ένα μη γραμμικό μοντέλο προγραμματισμού μπορεί να μορφοποιηθεί για αυτό το πρόβλημα, ως ακολούθως. Ας υποθέσουμε ότι n αποθέματα (τίτλοι), εξετάζονται για ένταξη στο χαρτοφυλάκιο, και έστω ότι οι μεταβλητές απόφασης x_j $(j = 1,2,...,n)$ είναι ο αριθμός των μετοχών του αποθέματος j που θα συμπεριληφθούν. Έστω, μ_j και σ_{jj} η (κατ 'εκτίμηση) *μέση τιμή* και *διασπορά*, αντίστοιχα, της επιστροφής για κάθε μετοχή του αποθέματος j, όπου σ_{jj} μετρά τον κίνδυνο αυτού του αποθέματος. Για $i = 1, 2,. . . , n$ $(i \neq j)$, έστω σ_{ij} η *συνδιακύμανση* της επιστροφής για κάθε μερίδιο της μετοχής i και j. (Επειδή, θα ήταν δύσκολο να εκτιμηθούν όλες οι τιμές σ_{ij}, η συνήθης προσέγγιση είναι να γίνουν ορισμένες υποθέσεις για τη συμπεριφορά της αγοράς, που θα μας επιτρέψουν να υπολογίσουμε το σ_{ij} απευθείας από τα σ_{ii} και σ_{jj}.) Στη συνέχεια, η αναμενόμενη τιμή *R(x)* και η διασπορά *V(x)* της συνολικής απόδοσης από το σύνολο του χαρτοφυλακίου είναι:

$$R(x) = \sum_{j=1}^{n} \mu_j x_j$$

και

$$V(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j,$$

όπου *V(x)* μετράει τον κίνδυνο που σχετίζεται με το χαρτοφυλάκιο. Ένας τρόπος για να εξετάσουμε τη σχέση ανάμεσα σε αυτούς τους δύο παράγοντες είναι να χρησιμοποιήσουμε το *V(x)* ως αντικειμενική συνάρτηση που πρέπει να ελαχιστοποιηθεί και στη συνέχεια, να επιβάλουμε τον περιορισμό ότι το *R(x)* δεν πρέπει να είναι μικρότερο από την ελάχιστη αποδεκτή αναμενόμενη απόδοση. Το πλήρες μοντέλο μη γραμμικού προγραμματισμού τότε θα είναι:

Ελαχιστοποίησε: $V(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sigma_{ij} x_i x_j$

Υπό τους περιορισμούς:

$$\sum_{j=1}^{n} \mu_j x_j \geq L$$

$$\sum_{j=1}^{n} P_j x_j \leq B$$

Και

$$x_j \geq 0, \text{ για } j = 1,2,\ldots,n,$$

όπου L η ελάχιστη αποδεκτή αναμενόμενη απόδοση, P_j η τιμή τού μερίσματος της μετοχής j και B το ποσό χρημάτων που προϋπολογίστηκαν για το χαρτοφυλάκιο.

Ένα μειονέκτημα αυτής της μορφοποίησης είναι το ότι, είναι σχετικά δύσκολο να επιλέξουμε μια κατάλληλη τιμή για το L για την απόκτηση του καλύτερου συμβιβασμού μεταξύ *R(x)* και *V(x)*. Ως εκ τούτου, παρά να σταματάμε με μία επιλογή τού L, είναι σύνηθες να χρησιμοποιούμε μία *παραμετρική* προσέγγιση (μη γραμμικού) προγραμματισμού, για να δημιουργήσουμε τη βέλτιστη λύση ως συνάρτηση του L, σε μία ευρεία περιοχή τιμών τού L. Το επόμενο βήμα είναι να εξετάσουμε τις τιμές των *R(x)* και *V(x)* για τις λύσεις, που είναι βέλτιστες για κάποια τιμή του L και στη συνέχεια να επιλέξουμε τη λύση που φαίνεται να δίνει τον καλύτερο συμβιβασμό μεταξύ αυτών των δύο ποσοτήτων. Αυτή η διαδικασία συχνά αναφέρεται ως δημιουργία των λύσεων για την *περιοχή αποδοχής* της δισδιάστατης γραφικής παράστασης των σημείων *(R(x), V(x))* των εφικτών *x*. Ο λόγος είναι ότι το σημείο *(R(x), V(x))* ενός βέλτιστου *x* (για κάποιο *L*) βρίσκεται στα σύνορα (όρια) των εφικτών σημείων. Επιπλέον, κάθε βέλτιστο *x* είναι *αποδοτικό*, με την έννοια δεν υπάρχει άλλη εφικτή λύση που να είναι τουλάχιστον εξίσου καλή με μία μέτρηση (*R* ή *V*) και αυστηρά καλύτερη από την άλλη μέτρηση (μικρότερο *V* ή μεγαλύτερο *R*).

2.4 Γραφική απεικόνιση προβλημάτων μη γραμμικού προγραμματισμού

Όταν ένα μη γραμμικό πρόβλημα προγραμματισμού έχει μόνο μία ή δύο μεταβλητές, μπορεί να παρασταθεί γραφικά από το πρόβλημα Wyndor Glass Co., για γραμμικό προγραμματισμό, που θα παρουσιαστεί παρακάτω. Επειδή, μια τέτοια γραφική παράσταση δίνει σημαντικές πληροφορίες για τις ιδιότητες των βέλτιστων λύσεων για γραμμικό και μη γραμμικό προγραμματισμό, ας δούμε μερικά παραδείγματα. Για να υπογραμμίσουμε τη διαφορά μεταξύ γραμμικού και μη γραμμικού προγραμματισμού, θα πρέπει να χρησιμοποιήσουμε κάποιες *μη γραμμικές* παραλλαγές του προβλήματος Wyndor Glass Co., το οποίο είναι το εξής:

Μεγιστοποίησε: $Z = 3x_1 + 5x_2$,

υπό τους περιορισμούς

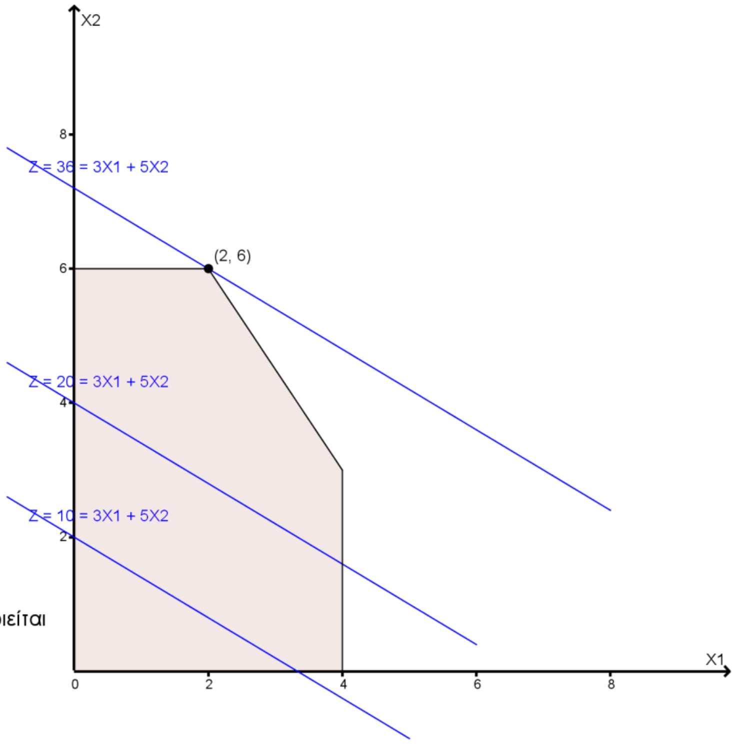
$x_1 \leq 4$

$2x_2 \leq 12$

$3x_1 + 2x_2 \leq 18$

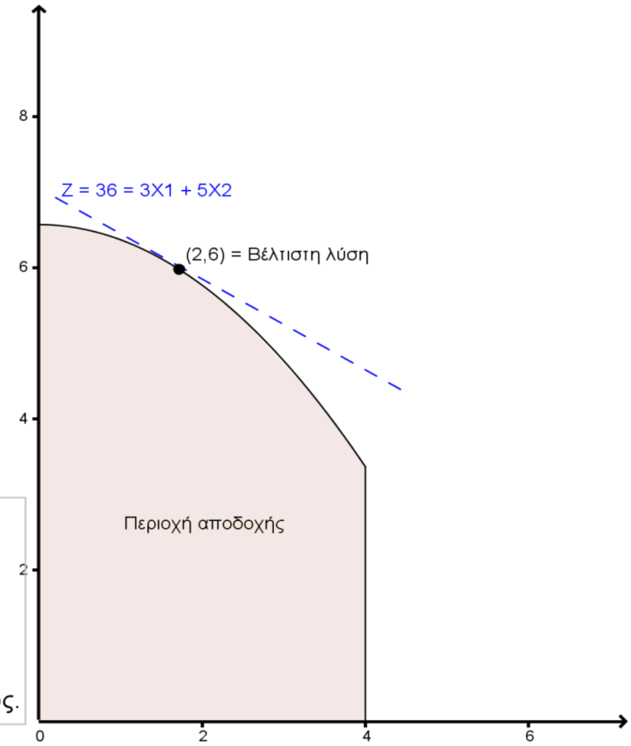
και

$x_1 \geq 0, x_2 \geq 0$



Σχήμα 2.4.1
Στην θέση (X1,X2) = (2,6) μεγιστοποιείται η αντικειμενική συνάρτηση.

Το Σχήμα 2.4.2 δείχνει τι συμβαίνει σε αυτό το πρόβλημα, αν οι μόνες αλλαγές στο υπόδειγμα είναι ότι τόσο η δεύτερη όσο και η τρίτη συνάρτηση περιορισμών αντικαθίστανται από τον ενιαίο μη γραμμικό περιορισμό $9{x_1}^2 + 5{x_2}^2 \leq 216$. Συγκρίνουμε το Σχ.2.4.2 με το Σχ.2.4.1. Η βέλτιστη λύση εξακολουθεί να είναι η $(x_1, x_2) = (2,6)$. Επιπλέον, εξακολουθεί να βρίσκεται στο σύνορο της εφικτής περιοχής. Ωστόσο, δεν είναι μια εφικτή λύση σε γωνιακό σημείο (corner-point feasible solution, CPF). Η βέλτιστη λύση θα μπορούσε να ήταν μια CPF λύση με μία διαφορετική αντικειμενική συνάρτηση (έλεγξε $Z = 3x_1 + x_2$), αλλά το γεγονός ότι δεν χρειάζεται να είναι μία τέτοια λύση σημαίνει ότι δεν έχουμε πλέον την τεράστια απλοποίηση, που χρησιμοποιείται στο γραμμικό προγραμματισμό, του περιορισμού της αναζήτησης για μία βέλτιστη λύση μόνο σε CPF λύσεις.

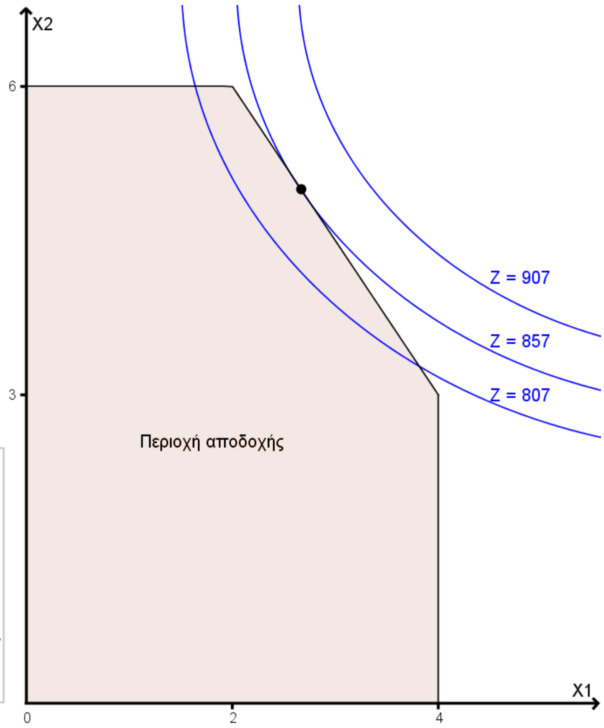


Σχήμα 2.4.2
Το πρόβλημα Wyndor Glass Co.
με τον μη γραμμικό περιορισμό
9X1^2 + 5X2^2 <= 216
στη θέση του δεύτερου και τρίτου
περιορισμού του αρχικού προβλήματος.

Τώρα, ας υποθέσουμε ότι οι γραμμικοί περιορισμοί του αρχικού προβλήματος παρέμειναν αμετάβλητοι, αλλά η αντικειμενική συνάρτηση γίνεται μη γραμμική. Για παράδειγμα, εάν:

$$Z = 126x_1 - 9x_1^2 + 182\, x_2 - 13\, x_2^2$$

τότε η γραφική παράσταση στο Σχ.2.4.3 δείχνει ότι η βέλτιστη λύση είναι $(x_1, x_2) = (\frac{8}{3}, 5)$, η οποία πάλι βρίσκεται στο σύνορο της εφικτής περιοχής. (Η τιμή τού *Z* για τη βέλτιστη λύση είναι *Z* = *857*. Έτσι, το Σχ.2.4.3 απεικονίζει το γεγονός ότι η θέση όλων των σημείων με *Z* = *857* τέμνει την εφικτή περιοχή ακριβώς σε αυτό το σημείο, ενώ η θέση των σημείων με οποιοδήποτε μεγαλύτερο *Z* δεν τέμνει κάπου την εφικτή περιοχή).

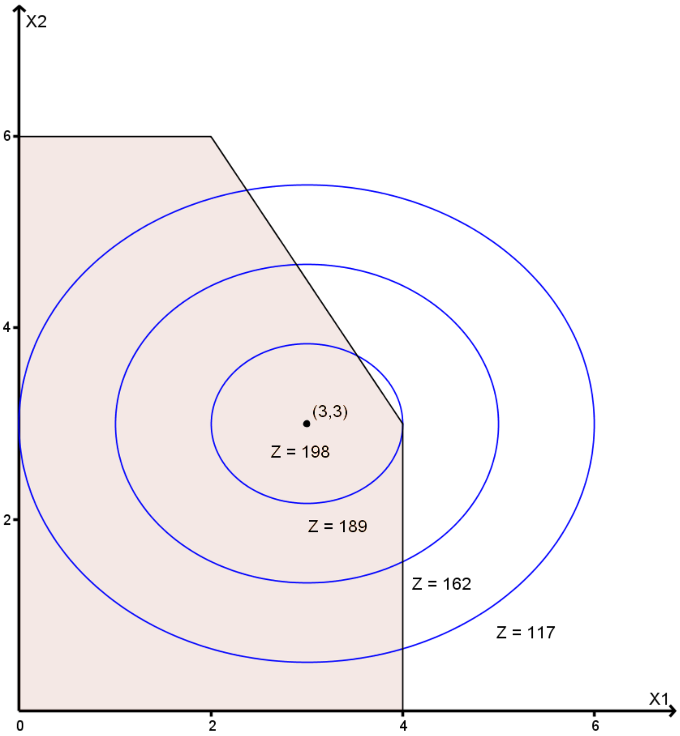


Σχήμα 2.4.3
Το πρόβλημα Wyndor Glass Co.
με την αρχική περιοχή αποδοχής,
αλλά με τη μη γραμμική αντικειμενική συνάρτηση Z = 126 X1 - 9 X1^2 + 182 X2 - 13 X2^2 στη θέση της αρχικής αντικειμενικής συνάρτησης.

Από την άλλη πλευρά, εάν:

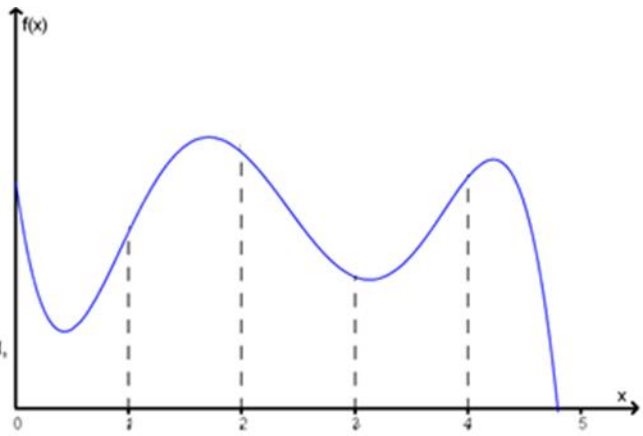
$$Z = 54x_1 - 9x_1^2 + 78x_2 - 13x_2^2,$$

τότε το Σχ.2.4.4 απεικονίζει ότι η βέλτιστη λύση, όπως αποδεικνύεται, είναι $(x_1, x_2) = (3,3)$, η οποία βρίσκεται *εντός* των ορίων της εφικτής περιοχής. (Μπορούμε να ελέγξουμε ότι η λύση αυτή είναι βέλτιστη με χρήση λογισμού, για να την εκλάβουμε ως το ολικό μέγιστο, χωρίς περιορισμούς. Επειδή, πληροί, επίσης, τους περιορισμούς, πρέπει να είναι η βέλτιστη, για το πρόβλημα περιορισμών.) Ως εκ τούτου, ένας γενικός αλγόριθμος, για την επίλυση παρόμοιων προβλημάτων, πρέπει να εξετάσει όλες τις λύσεις στην εφικτή περιοχή και όχι μόνο εκείνες που αφορούν το σύνορο.



Σχήμα 2.4.4
Το πρόβλημα Wyndor Glass Co.
με την ίδια περιοχή αποδοχής,
αλλά με μία άλλη μη γραμμική
αντικειμενική συνάρτηση, την
Z = 54X1 - 9X1^2 + 78X2 - 13X2^2,
να αντικαθιστά την αρχική αντικειμενική
συνάρτηση.

Μια άλλη περιπλοκή, που προκύπτει σε μη γραμμικό προγραμματισμό, είναι ότι ένα τοπικό μέγιστο δεν είναι κατ’ ανάγκη και *ολικό* μέγιστο (συνολική βέλτιστη λύση). Για παράδειγμα, ας θεωρήσουμε την συνάρτηση μιας μεταβλητής, που παριστάνεται γραφικά στο Σχ.2.4.5. Στο διάστημα $0 \leq x \leq 5$, η συνάρτηση αυτή έχει τρία τοπικά μέγιστα ($x = 0$, $x = 1.8$, και $x = 4.2$), αλλά μόνο ένα από αυτά ($x = 1.8$) είναι *ολικό μέγιστο*. (Ομοίως, υπάρχουν τοπικά ελάχιστα στο $x = 0.5$, 3, και 5, αλλά μόνο το $x = 5$ είναι *ολικό ελάχιστο*.)



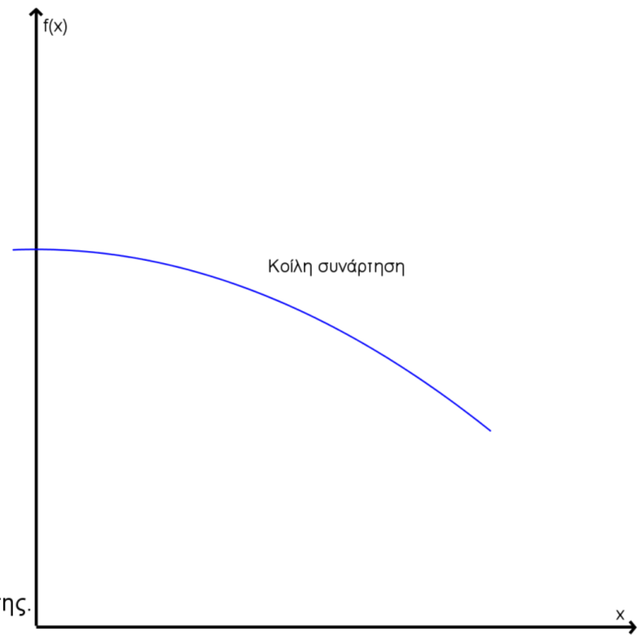
Σχήμα 2.4.5
Συνάρτηση με 3 τοπικά μέγιστα,
αλλά μόνο το ένα είναι ολικό.

Οι αλγόριθμοι μη γραμμικού προγραμματισμού, γενικά, δεν είναι σε θέση να διακρίνουν μεταξύ ενός τοπικού και ενός ολικού μεγίστου (εκτός από την εύρεση ενός άλλου *καλύτερου* τοπικού μεγίστου). Ως εκ τούτου, καθίσταται σημαντικό να γνωρίζουμε τις συνθήκες κάτω από τις οποίες κάθε τοπικό μέγιστο είναι *εγγυημένα* ένα ολικό μέγιστο πάνω στην εφικτή περιοχή. Είναι γνωστό από τον λογισμό, ότι όταν μεγιστοποιούμε μια συνηθισμένη (διπλά διαφορίσιμη) συνάρτηση μιας μεταβλητής $f(x)$, χωρίς κανέναν περιορισμό, η συνθήκη που πρέπει να ικανοποιείται είναι:

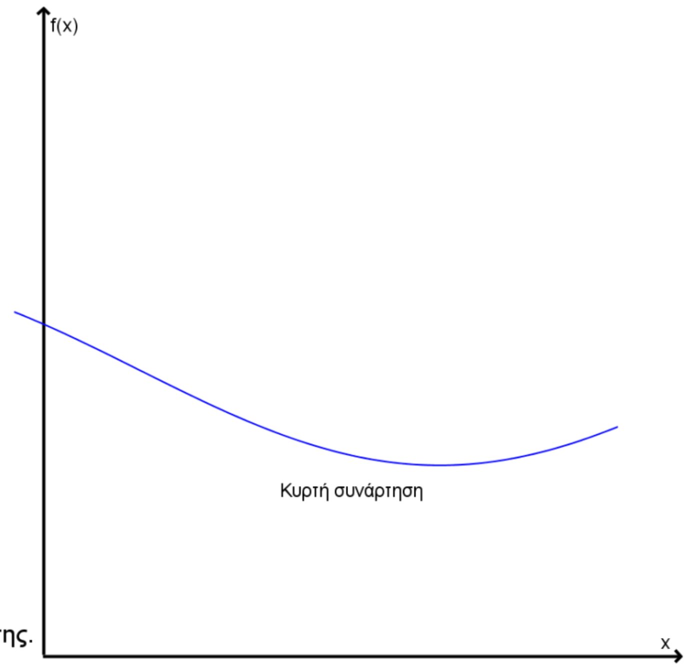
$$\frac{d^2f}{dx^2} \leq 0 \text{ για κάθε } x.$$

Μια τέτοια συνάρτηση που είναι πάντα ‘‘καμπτόμενη προς τα κάτω’’ (ή καθόλου καμπτόμενη) ονομάζεται **κοίλη** συνάρτηση. Ομοίως, εάν όπου $\leq$ αντικαταστήσουμε με $\geq$, έτσι ώστε η συνάρτηση να είναι πάντα ‘‘καμπτόμενη προς τα άνω’’ (ή καθόλου καμπτόμενη), ονομάζεται **κυρτή** συνάρτηση. (Έτσι, μια γραμμική συνάρτηση είναι και κοίλη και κυρτή.) Δείτε τα Σχ.2.4.6 και 2.4.7 για παραδείγματα. Στη συνέχεια, σημειώστε ότι το Σχ. 2.4.5 απεικονίζει μια συνάρτηση, που δεν είναι ούτε κοίλη ούτε κυρτή, επειδή εναλλάσσεται μεταξύ κάμψης προς τα άνω και προς τα κάτω.

Συναρτήσεις πολλών μεταβλητών μπορούν, επίσης, να χαρακτηριστούν ως κοίλες ή κυρτές, εάν είναι πάντοτε καμπύλες προς τα κάτω ή προς τα άνω, αντίστοιχα.



Σχήμα 2.4.6
Παράδειγμα κοίλης συνάρτησης.



Σχήμα 2.4.7
Παράδειγμα κυρτής συνάρτησης.

Ακολουθεί ένας βολικός τρόπος για τον έλεγχο αν μία συνάρτηση περισσότερων από δύο μεταβλητών, είναι κοίλη ή κυρτή, όταν η συνάρτηση αποτελείται από ένα *άθροισμα* μικρότερων συναρτήσεων με μόνο μία ή δύο μεταβλητές η καθεμιά. Εάν κάθε επιμέρους συνάρτηση είναι κοίλη, τότε η συνολική συνάρτηση είναι κοίλη. Παρομοίως, η συνολική συνάρτηση είναι κυρτή εάν κάθε επιμέρους συνάρτηση είναι κυρτή.

Για να το δούμε, εξετάζουμε τη συνάρτηση:

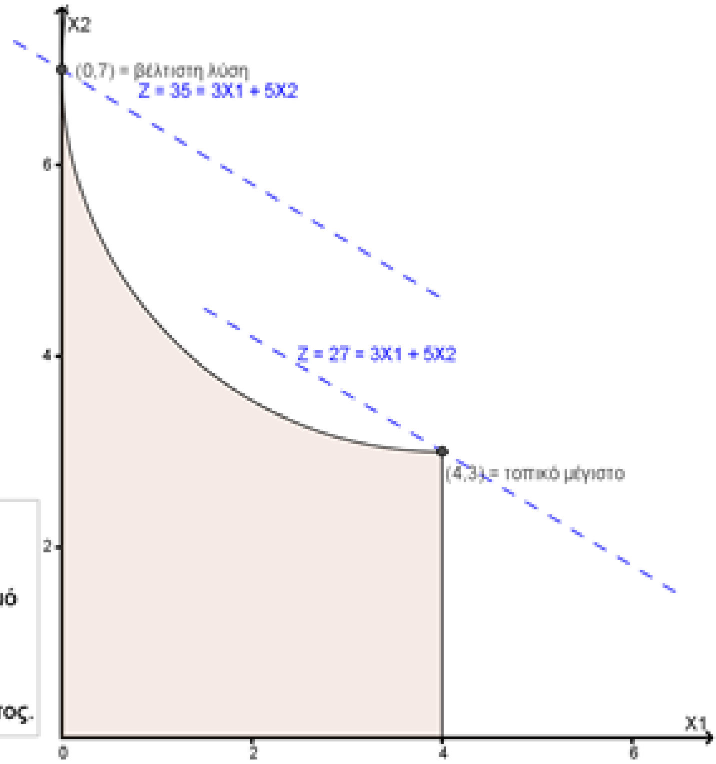
$$
\begin{aligned}
f(x_1, x_2, x_3) &= 4x_1 - x_1^2 - (x_2 - x_3)^2 \\
&= [4x_1 - x_1^2] + [-(x_2 - x_3)^2],
\end{aligned}
$$

που είναι το άθροισμα των δύο επιμέρους συναρτήσεων, που δίνονται σε αγκύλες. Η πρώτη συνάρτηση $4x_1 - x_1^2$ είναι μία συνάρτηση της απλής μεταβλητής x_1, έτσι ώστε να μπορεί να βρεθεί ότι είναι κοίλη παρατηρώντας ότι ο δεύτερος παράγοντάς της είναι αρνητικός. Η δεύτερη συνάρτηση $-(x_2 - x_3)^2$ είναι μια συνάρτηση μόνο των x_2 και x_3. Επειδή και οι δύο μικρότερες συναρτήσεις είναι κοίλες, η συνολική συνάρτηση $f(x_1, x_2, x_3)$ πρέπει να είναι κοίλη.

Αν ένα πρόβλημα μη γραμμικού προγραμματισμού δεν έχει περιορισμούς και η αντικειμενική συνάρτηση είναι *κοίλη*, τότε εξασφαλίζεται ότι ένα τοπικό μέγιστο είναι *ολικό*. (Ομοίως, όταν η αντικειμενική συνάρτηση είναι *κυρτή*, τότε εξασφαλίζεται ότι ένα τοπικό ελάχιστο είναι *ολικό*.) Εάν υπάρχουν περιορισμοί, τότε μία ακόμη συνθήκη θα παράσχει την εγγύηση αυτή, δηλαδή, ότι η *περιοχή αποδοχής* είναι ένα **κυρτό σύνολο**. Είναι γνωστό ότι ένα κυρτό σύνολο είναι ένα σύνολο από σημεία τέτοια ώστε, για κάθε ζεύγος σημείων τού συνόλου, ολόκληρο το τμήμα της γραμμής που ενώνει τα δύο αυτά σημεία ανήκει επίσης στο σύνολο. Έτσι, η περιοχή λύσεων για το αρχικό πρόβλημα Wyndor Glass Co. (βλέπε Σχ.2.4.3 ή 2.4.4) είναι κυρτό σύνολο. Στην πραγματικότητα, η περιοχή λύσεων για κάθε πρόβλημα γραμμικού προγραμματισμού είναι κυρτό σύνολο. Ομοίως, η εφικτή περιοχή στο Σχ.2.4.2 είναι κυρτό σύνολο.

Σε γενικές γραμμές, η περιοχή λύσεων για ένα πρόβλημα μη γραμμικού προγραμματισμού είναι κυρτό σύνολο, όταν όλες οι $g_i(x)$ [στους περιορισμούς $g_i(x) \leq b_i$] είναι κυρτές συναρτήσεις. Για το παράδειγμα του Σχ.2.4.2, και οι δύο $g_i(x)$ είναι κυρτές συναρτήσεις, αφού η $g_1(x) = x_1$ (μια γραμμική συνάρτηση είναι κοίλη και κυρτή) και η $g_2(x) = 9x_1^2 + 5x_2^2$ (οι $9x_1^2$ και $5x_2^2$ είναι κυρτές συναρτήσεις, οπότε το άθροισμά τους είναι μια κυρτή συνάρτηση). Οι δύο κυρτές $g_i(x)$ οδηγούν στο ότι η εφικτή περιοχή του Σχ.2.4.2 είναι κυρτό σύνολο.

Τώρα, ας δούμε τι συμβαίνει όταν, αντιθέτως, μόνο μία από τις $g_i(x)$ είναι κοίλη συνάρτηση. Συγκεκριμένα, ας υποθέσουμε ότι η μόνη αλλαγή που έγινε στο παράδειγμα του Σχ.2.4.2 είναι ότι ο μη γραμμικός περιορισμός του αντικαθίσταται από τον $8x_1 - x_1^2 + 14x_2 - x_2^2 \leq 49$. Κατά συνέπεια, η νέα συνάρτηση $g_2(x) = 8x_1 - x_1^2 + 14x_2 - x_2^2$ είναι κοίλη συνάρτηση, αφού τόσο η $8x_1 - x_1^2$ όσο και η $14x_2 - x_2^2$ η είναι κοίλες συναρτήσεις. Η νέα περιοχή λύσεων που φαίνεται στο Σχ.2.4.8 δεν είναι κυρτό σύνολο. Αυτό συμβαίνει διότι η περιοχή αποδοχής περιέχει ζεύγη σημείων, για παράδειγμα, (0, 7) και (4, 3), για τα οποία μέρος του τμήματος γραμμής που ενώνει αυτά τα δύο σημεία δεν ανήκει στην περιοχή αποδοχής. Κατά συνέπεια, δεν μπορούμε να εγγυηθούμε ότι ένα τοπικό μέγιστο είναι ολικό. Στην πραγματικότητα, αυτό το παράδειγμα έχει δύο τοπικά μέγιστα (0, 7) και (4, 3), αλλά μόνο το (0, 7) είναι ολικό μέγιστο.



Σχήμα 2.4.8
Το πρόβλημα Wyndor Glass Co.
με έναν άλλο μη γραμμικό περιορισμό
8X1 - X1^2 + 14X2 - X2^2 <= 49,
στη θέση του δεύτερου και τρίτου
περιορισμού του αρχικού προβλήματος.

Ως εκ τούτου, προκειμένου να διασφαλιστεί ότι ένα τοπικό μέγιστο είναι ολικό, για ένα πρόβλημα μη γραμμικού προγραμματισμού με περιορισμούς $g_i(x) \leq b_i (i = 1, 2, \ldots, m)$ και $x \geq 0$, η αντικειμενική συνάρτηση $f(x)$ πρέπει να είναι κοίλη και κάθε $g_i(x)$ πρέπει να είναι κυρτή συνάρτηση. Ένα τέτοιο πρόβλημα ονομάζεται *πρόβλημα κυρτού προγραμματισμού* και είναι μία από τις βασικές μορφές προβλημάτων μη γραμμικού προγραμματισμού, που θα συζητηθούν παρακάτω.

2.5 Είδη προβλημάτων μη γραμμικού προγραμματισμού

Προβλήματα μη γραμμικού προγραμματισμού εμφανίζονται σε πολλές διαφορετικές μορφές. Σε αντίθεση με τη μέθοδο Simplex για τον γραμμικό προγραμματισμό, δεν υπάρχει αλγόριθμος που να μπορεί να λύσει όλα αυτά τα διαφορετικά είδη προβλημάτων. Αντ' αυτού, έχουν αναπτυχθεί αλγόριθμοι για διάφορες κλάσεις (ειδικές μορφές) των προβλημάτων μη γραμμικού προγραμματισμού. Οι πιο σημαντικές κατηγορίες εισάγονται εν συντομία σε αυτή την ενότητα. Στις ενότητες που ακολουθούν στη συνέχεια, περιγράφεται πώς μπορούν να λυθούν κάποια προβλήματα αυτών των μορφών.

Βελτιστοποίηση χωρίς περιορισμούς

Εφόσον, τα προβλήματα βελτιστοποίησης χωρίς περιορισμούς δεν έχουν περιορισμούς, ο στόχος είναι απλά να:

Μεγιστοποιήσουμε το $f(\mathbf{x})$

επί *όλων* των τιμών του $\mathbf{x}=(x_1,x_2,\ldots,x_n)$. Απαραίτητη προϋπόθεση, ώστε μια συγκεκριμένη λύση $x = x^*$ να είναι η βέλτιστη, όταν η $f(x)$ είναι διαφορίσιμη συνάρτηση είναι:

$\frac{\partial f}{\partial x_j} = 0$ στο $\mathbf{x} = \mathbf{x^*}$, για $j = 1,2,\ldots,n$.

Όταν η $f(x)$ είναι κοίλη συνάρτηση, είναι μία επαρκής συνθήκη, ώστε στη συνέχεια η επίλυση για x^* να καταλήγει στην επίλυση του συστήματος των n εξισώσεων, που παίρνουμε από τη συνθήκη οι n μερικές παράγωγοι να ισούνται με μηδέν. Δυστυχώς, για μη γραμμικές συναρτήσεις $f(x)$, οι εξισώσεις αυτές συχνά θα είναι μη γραμμικές, οπότε είναι απίθανο να είμαστε σε θέση να επιτύχουμε την ταυτόχρονη αναλυτική τους λύση. Οι ενότητες 2.6 και 2.7 περιγράφουν *αλγοριθμικές διαδικασίες αναζήτησης* για την εύρεση x^*, πρώτα για $n = 1$ και στη συνέχεια, για $n > 1$. Αυτές οι διαδικασίες διαδραματίζουν, επίσης, σημαντικό ρόλο στην επίλυση πολλών από τις μορφές του προβλήματος, που περιγράφεται στη συνέχεια, όπου υπάρχουν περιορισμοί. Ο λόγος είναι ότι πολλοί αλγόριθμοι για προβλήματα *με περιορισμούς* έχουν σχεδιαστεί, έτσι ώστε να μπορούν να επικεντρωθούν σε μια εκδοχή του προβλήματος *χωρίς περιορισμούς*, κατά την διάρκεια ενός τμήματος της κάθε επανάληψης.

Όταν μία μεταβλητή x_j έχει έναν περιορισμό μη αρνητικότητας $x_j \geq 0$, η προηγούμενη αναγκαία και (ίσως) επαρκής συνθήκη γίνεται:

$$\frac{\partial f}{\partial x_j}\begin{Bmatrix} \leq 0 \; \sigma\tau o \; x = x^*, \alpha\nu \; x_j^* = 0 \\ = 0 \; \sigma\tau o \; x = x^*, \alpha\nu \; x_j^* > 0 \end{Bmatrix}$$

για κάθε τέτοιο j. Αυτή η συνθήκη απεικονίζεται στο Σχ.2.5.1, όπου η βέλτιστη λύση για ένα πρόβλημα με μία μόνο μεταβλητή είναι στο $x = 0$, παρόλο που η παράγωγος είναι αρνητική, μη μηδενική. Επειδή, αυτό το παράδειγμα αναφέρεται στη μεγιστοποίηση μίας κοίλης συνάρτησης, υποκείμενης σε περιορισμό μη αρνητικότητας, με την παράγωγο μικρότερη ή ίση με 0, στο $x = 0$ είναι τόσο αναγκαία και ικανή προϋπόθεση, για να είναι βέλτιστη λύση η $x = 0$.

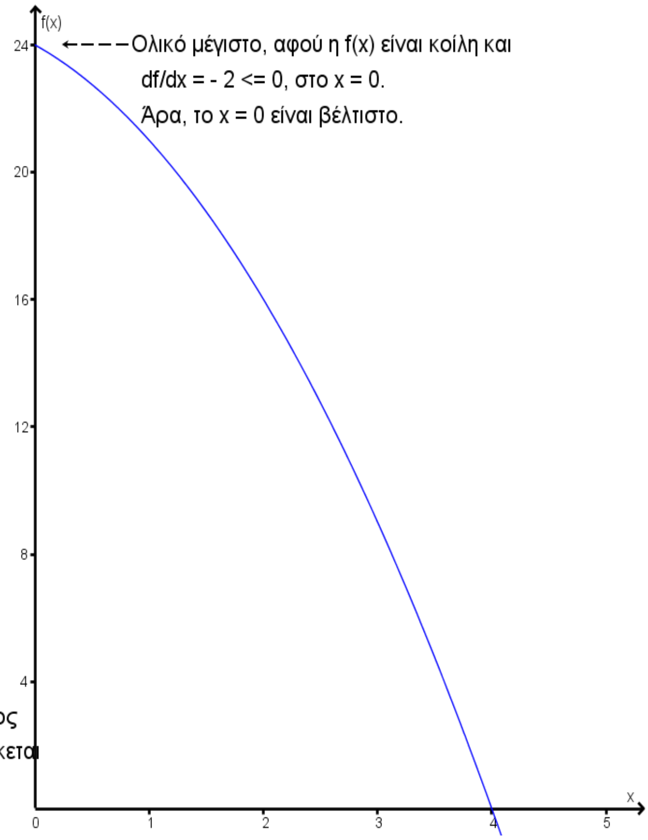
Ένα πρόβλημα, που έχει κάποιους περιορισμούς μη αρνητικότητας, αλλά δεν έχει συναρτήσεις περιορισμών είναι μία ειδική περίπτωση ($m = 0$) της επόμενης τάξης των προβλημάτων.

Γραμμικά Περιορισμένη Βελτιστοποίηση

Τα γραμμικά περιορισμένα προβλήματα βελτιστοποίησης, που χαρακτηρίζονται από περιορισμούς, που ταιριάζουν απόλυτα με το γραμμικό προγραμματισμό, έτσι ώστε όλοι οι $g_i(x)$ περιορισμοί να είναι γραμμικοί, αλλά η αντικειμενική συνάρτηση $f(x)$ μη γραμμική. Το πρόβλημα απλοποιείται σημαντικά, αν έχουμε μόνο μία μη γραμμική συνάρτηση να λάβουμε υπόψη, με εφικτή περιοχή γραμμικού προγραμματισμού. Έχει αναπτυχθεί ένας αριθμός ειδικών αλγορίθμων βασισμένων στην επέκταση της μεθόδου *Simplex*, για να εξετασθεί η μη γραμμική αντικειμενική συνάρτηση. Μια σημαντική ειδική περίπτωση, η οποία θα εξετασθεί παρακάτω, είναι ο τετραγωνικός προγραμματισμός.

Τετραγωνικός Προγραμματισμός

Τα προβλήματα τετραγωνικού προγραμματισμού έχουν επίσης γραμμικούς περιορισμούς, αλλά η αντικειμενική συνάρτηση $f(x)$ πρέπει να είναι τετραγωνική. Έτσι, η μόνη διαφορά μεταξύ ενός τέτοιου προβλήματος και ενός προβλήματος γραμμικού προγραμματισμού είναι ότι ορισμένοι από τους όρους στην αντικειμενική συνάρτηση περιλαμβάνουν το *τετράγωνο* μιας μεταβλητής ή το *γινόμενο* δύο μεταβλητών.



Σχήμα 2.5.1
Ένα παράδειγμα, που δείχνει πώς μία βέλτιστη λύση μπορεί είναι ένα σημείο, όπου η παράγωγος είναι αρνητική ή μηδενική, επειδή το σημείο βρίσκεται στο σύνορο του περιορισμού μη αρνητικότητας.

Το σχήμα απεικονίζει το πρόβλημα:

Μεγιστοποίησε: $f(x) = 24 - 2x - x^2$,

υπό τον περιορισμό $x \geq 0$.

Πολλοί αλγόριθμοι έχουν αναπτυχθεί για την περίπτωση αυτή, υπό την πρόσθετη προϋπόθεση ότι *f(x)* είναι κοίλη συνάρτηση.

Ο τετραγωνικός προγραμματισμός είναι πολύ σημαντικός, κυρίως επειδή τέτοιες μορφές προκύπτουν φυσικά σε πολλές εφαρμογές. Για παράδειγμα, το πρόβλημα της επιλογής χαρτοφυλακίου με ριψοκίνδυνους τίτλους, που αναφέρεται στην ενότητα 2.3 ταιριάζει σε αυτή τη μορφή. Ωστόσο, ένας άλλος βασικός λόγος, για τον οποίο είναι σημαντικός, είναι ότι μια κοινή προσέγγιση, για την επίλυση γραμμικά περιορισμένων προβλημάτων βελτιστοποίησης, είναι να λύσουμε μια σειρά προσεγγίσεων τετραγωνικού προγραμματισμού.

Κυρτός προγραμματισμός

Ο *κυρτός προγραμματισμός* καλύπτει μια ευρεία κατηγορία των προβλημάτων, που περιλαμβάνει στην πραγματικότητα ως ειδικές περιπτώσεις όλους τους προηγούμενους τύπους, όταν η $f(x)$ είναι κοίλη συνάρτηση. Οι υποθέσεις είναι οι εξής:

1. Η $f(x)$ είναι κοίλη συνάρτηση.

2. Κάθε $g_i(x)$ είναι κυρτή συνάρτηση.

Όπως συζητήθηκε στο τέλος της ενότητας 2.4, οι υποθέσεις αυτές είναι αρκετές για να εξασφαλίσουν ότι ένα τοπικό μέγιστο είναι ολικό μέγιστο. Θα δούμε στην ενότητα 4.1 ότι οι αναγκαίες και ικανές συνθήκες για μια τέτοια βέλτιστη λύση είναι μια φυσική γενίκευση των συνθηκών βελτιστοποίησης χωρίς περιορισμούς και επέκτασή τους, ώστε να συμπεριλάβουν περιορισμούς μη αρνητικότητας. Στη συνέχεια η ενότητα 4.4 περιγράφει αλγοριθμικές προσεγγίσεις για την επίλυση προβλημάτων κυρτού προγραμματισμού.

Διαχωρίσιμος Προγραμματισμός

Διαχωρίσιμος προγραμματισμός: είναι μια ειδική περίπτωση του κυρτού προγραμματισμού, όπου η μία επιπλέον υπόθεση είναι ότι:

3. Όλες οι $f(x)$ και $g_i(x)$ είναι διαχωρίσιμες συναρτήσεις.

Διαχωρίσιμη: είναι μια συνάρτηση, που κάθε της όρος περιλαμβάνει μία μόνο μεταβλητή, έτσι ώστε να μπορεί να διαχωριστεί σε ένα άθροισμα συναρτήσεων των επιμέρους μεταβλητών. Για παράδειγμα, εάν $f(x)$ είναι μια διαχωρίσιμη συνάρτηση, μπορεί να εκφραστεί ως:

$$f(x) = \sum_{j=1}^{n} f_j(x_j),$$

όπου κάθε συνάρτηση $f_j(x_j)$ περιλαμβάνει μόνο τους όρους που αφορούν το x_j. Σύμφωνα με την ορολογία του γραμμικού προγραμματισμού, τα προβλήματα διαχωρίσιμου προγραμματισμού ικανοποιούν την υπόθεση της προσθετικότητας, αλλά όχι την υπόθεση της αναλογικότητας (για μη γραμμικές συναρτήσεις).

Για παράδειγμα, η αντικειμενική συνάρτηση που εξετάσαμε στο Σχ. 2.4.3:

$$f(x_{1,}x_2) = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2$$

είναι μία διαχωρίσιμη συνάρτηση, διότι μπορεί να γραφτεί ως:

$$f(x_{1,}x_2) = f_1(x_1) + f_2(x_2)$$

όπου $f_1(x_1) = 126x_1 - 9x_1^2$ και $f_2(x_2) = 182x_2 - 13x_2^2$ είναι η καθεμιά συνάρτηση μίας μεταβλητής – x_1 και x_2, αντίστοιχα. Με την ίδια λογική, μπορούμε να επιβεβαιώσουμε ότι η αντικειμενική συνάρτηση του Σχ.2.4.4 είναι επίσης διαχωρίσιμη.

Είναι σημαντικό να διακρίνουμε τα προβλήματα διαχωρίσιμου προγραμματισμού από άλλα προβλήματα κυρτού προγραμματισμού, διότι κάθε τέτοιο πρόβλημα μπορεί να προσεγγιστεί με ακρίβεια από ένα πρόβλημα γραμμικού προγραμματισμού, έτσι ώστε να μπορεί να χρησιμοποιείται η εξαιρετικά αποτελεσματική μέθοδος Simplex. Αυτή η προσέγγιση περιγράφεται στην ενότητα 4.3. (Στην οποία, για λόγους απλότητας, θα επικεντρωθούμε στην περίπτωση *γραμμικά περιορισμένων* προβλημάτων, όπου η ειδική προσέγγιση είναι απαραίτητη μόνο για την αντικειμενική συνάρτηση.)

Μη κυρτός Προγραμματισμός

Ο *μη κυρτός προγραμματισμός* περιλαμβάνει όλα τα μη γραμμικά προβλήματα προγραμματισμού, που δεν πληρούν τις προϋποθέσεις του κυρτού προγραμματισμού. Τώρα, ακόμα και αν είμαστε σε θέση να βρούμε ένα *τοπικό μέγιστο*, δεν υπάρχει καμία διαβεβαίωση ότι θα είναι και *ολικό μέγιστο*. Ως εκ τούτου, δεν υπάρχει αλγόριθμος που να εγγυάται την εύρεση μιας βέλτιστης λύσης για όλα αυτά τα προβλήματα. Ωστόσο, υπάρχουν ορισμένοι αλγόριθμοι που είναι σχετικά καλά προσαρμοσμένοι για την εύρεση τοπικών μεγίστων, ιδιαίτερα όταν οι μορφή των συναρτήσεων δεν αποκλίνει πολύ από αυτή που πρέπει να έχουν τα προβλήματα κυρτού προγραμματισμού. Ένας τέτοιος αλγόριθμος παρουσιάζεται στην ενότητα 4.5.

Ωστόσο, ορισμένοι τύποι προβλημάτων μη κυρτού προγραμματισμού μπορούν να επιλυθούν, χωρίς μεγάλη δυσκολία, με ειδικές μεθόδους. Δύο ιδιαίτερα σημαντικοί τύποι συζητούνται, εν συντομία, στη συνέχεια.

Γεωμετρικός Προγραμματισμός

Όταν εφαρμόζουμε μη γραμμικό προγραμματισμό σε προβλήματα τεχνικού σχεδιασμού, η αντικειμενική συνάρτηση και οι συναρτήσεις περιορισμών συχνά λαμβάνουν τη μορφή:

$$g(x) = \sum_{i=1}^{N} c_i\, P_i(x),$$

όπου

$$P_i(x) = x_1^{a_{i1}} x_2^{a_{i2}} \ldots x_n^{a_{in}}, \text{ για } i = 1, 2, \ldots, n.$$

Σε τέτοιες περιπτώσεις, τα c_i και α_{ij} αντιπροσωπεύουν φυσικές σταθερές, και τα x_j είναι μεταβλητές σχεδιασμού. Αυτές οι συναρτήσεις δεν είναι γενικά ούτε κυρτές ούτε κοίλες και έτσι οι τεχνικές του κυρτού προγραμματισμού δεν μπορούν να εφαρμοστούν απευθείας στα προβλήματα γεωμετρικού προγραμματισμού. Ωστόσο, υπάρχει μία σημαντική περίπτωση, όπου το πρόβλημα μπορεί να μετατραπεί σε ένα ισοδύναμο πρόβλημα κυρτού προγραμματισμού. Αυτή η περίπτωση είναι όταν όλοι οι συντελεστές c_i σε κάθε συνάρτηση είναι απολύτως θετικοί, έτσι ώστε οι συναρτήσεις να είναι *ολικά θετικά πολυώνυμα* -(τώρα ονομάζονται *posynomials*)- και η αντικειμενική συνάρτηση θα πρέπει να ελαχιστοποιηθεί. Το ισοδύναμο πρόβλημα κυρτού προγραμματισμού με μεταβλητές απόφασης $y_1, y_2, \ldots, y_n$ λαμβάνεται στη συνέχεια, θέτοντας

$$x_j = e^{y_j}, \text{ για } i = 1, 2, \ldots, n$$

σε όλο το αρχικό μοντέλο, έτσι ώστε να μπορεί να εφαρμοστεί ένας αλγόριθμος κυρτού προγραμματισμού. Εναλλακτικές διαδικασίες έχουν, επίσης, αναπτυχθεί για την επίλυση των προβλημάτων posynomial προγραμματισμού, καθώς και για τα προβλήματα γεωμετρικού προγραμματισμού άλλων τύπων.

Κλασματικός Προγραμματισμός

Ας υποθέσουμε ότι η αντικειμενική συνάρτηση είναι υπό τη μορφή ενός κλάσματος, δηλαδή ο λόγος των δύο συναρτήσεων,

Μεγιστοποίησε $f(x) = \frac{f_1(x)}{f_2(x)}$.

Τέτοια προβλήματα *κλασματικού προγραμματισμού* προκύπτουν, π.χ., όταν κάποιος μεγιστοποιεί τον λόγο της απόδοσης προσώπων-ωρών που δαπανήθηκαν (παραγωγικότητα), και του κέρδους για το κεφάλαιο που δαπανήθηκε (ποσοστό απόδοσης), ή την αναμενόμενη τιμή στην τυπική απόκλιση κάποιου μέτρου της απόδοσης για ένα χαρτοφυλάκιο επενδύσεων (επιστροφή/ρίσκο). Ορισμένες ειδικές διαδικασίες λύσης έχουν αναπτυχθεί για ορισμένες μορφές $f_1(x)$ και $f_2(x)$.

Όταν μπορεί να γίνει, η πιο απλή προσέγγιση για την επίλυση ενός προβλήματος κλασματικού προγραμματισμού είναι να μετατραπεί σε ένα ισοδύναμο πρόβλημα του βασικού τύπου για το οποίο υπάρχουν αποτελεσματικές διαδικασίες λύσης. Για επεξήγηση, ας υποθέσουμε ότι η $f(x)$ είναι *γραμμική μορφή κλασματικού προγραμματισμού*.

$$f(x) = \frac{cx + c_0}{dx + d_0}$$

όπου τα c και d είναι διανύσματα, το x είναι διάνυσμα-στήλη και τα c_0 και d_0 είναι βαθμωτά μεγέθη. Επίσης, υποθέτουμε ότι οι συναρτήσεις περιορισμών $g_i(x)$ είναι γραμμικές, έτσι ώστε οι περιορισμοί σε μορφή πίνακα να είναι $Ax \leq b$ και $x \geq 0$.

Κάτω από ήπιες πρόσθετες παραδοχές, μπορούμε να μετατρέψουμε το πρόβλημα σε ένα ισοδύναμο πρόβλημα γραμμικού προγραμματισμού, θέτοντας:

$y = \frac{x}{dx+d_0}$ και $t = \frac{1}{dx+d_0}$,

όπου $x = y/t$. Έτσι, έχουμε ως αποτέλεσμα το εξής πρόβλημα:

Μεγιστοποίησε $Z = cy + c_0t$,

υπό τους περιορισμούς:

$Ay - bt \leq 0$,

$dy + d_0 t = 1,$

και

$y \geq 0, t \geq 0,$

το οποίο μπορεί να λυθεί με τη μέθοδο *Simplex*. Γενικότερα, το ίδιο είδος μετασχηματισμού μπορεί να χρησιμοποιηθεί για να μετατρέψει ένα πρόβλημα κλασματικού προγραμματισμού με κοίλη $f_1(x)$, κυρτή $f_2(x)$, και κυρτές $g_i(x)$ σε ένα ισοδύναμο πρόβλημα κυρτού προγραμματισμού.

Το Πρόβλημα Συμπληρωματικότητας

Όταν θα ασχοληθούμε με προβλήματα τετραγωνικού προγραμματισμού στην ενότητα 4.2, θα δούμε ένα παράδειγμα του πώς η επίλυση ορισμένων προβλημάτων μη γραμμικού προγραμματισμού μπορεί αναχθεί στην επίλυση του προβλήματος της συμπληρωματικότητας. Με δεδομένες τις μεταβλητές $w_1, w_2, \ldots, w_p$ και $z_1, z_2, \ldots, z_p$, το **πρόβλημα συμπληρωματικότητας** είναι να βρούμε μια *εφικτή* λύση για το σύνολο των παρακάτω περιορισμών:

$w = F(z), w \geq 0, z \geq 0$

η οποία ικανοποιεί επίσης τον **συμπληρωματικό περιορισμό:**

$$w^T z = 0.$$

Εδώ, τα w και z είναι διανύσματα στήλες, F είναι μια συγκεκριμένη διανυσματική συνάρτηση, και ο εκθέτης T υποδηλώνει τον ανάστροφο. Το πρόβλημα δεν έχει καμία αντικειμενική συνάρτηση, τόσο τεχνικά δεν είναι ένα πλήρες μη γραμμικό πρόβλημα προγραμματισμού. Ονομάζεται το πρόβλημα της συμπληρωματικότητας, λόγω των συμπληρωματικών σχέσεων που είτε:

$w_i = 0$, είτε $z_i = 0$, για κάθε $i = 1, 2, \ldots, p$.

Μια σημαντική ειδική περίπτωση είναι το γραμμικό πρόβλημα συμπληρωματικότητας, όπου $F(z) = q + Mz$,

όπου το q είναι ένα δεδομένο διάνυσμα στήλη και το M είναι ένας δεδομένος p x p πίνακας. Αποδοτικοί αλγόριθμοι έχουν αναπτυχθεί για την επίλυση αυτού του προβλήματος υπό κατάλληλες υποθέσεις, σχετικά με τις ιδιότητες του πίνακα M. Ένας τύπος περιλαμβάνει την αναδιάταξη από μία βασική εφικτή λύση στην επόμενη, πολύ παρόμοια με τη μέθοδο Simplex για τον γραμμικό προγραμματισμό.

Εκτός από τις εφαρμογές τους στον μη γραμμικό προγραμματισμό, τα προβλήματα συμπληρωματικότητας έχουν εφαρμογές στην θεωρία παιγνίων, τα προβλήματα οικονομικής ισορροπίας, και τα προβλήματα μηχανικής ισορροπίας.

Κεφάλαιο 3: Βελτιστοποίηση χωρίς περιορισμούς

3.1 Βελτιστοποίηση προβλημάτων μιας μεταβλητής, χωρίς περιορισμούς

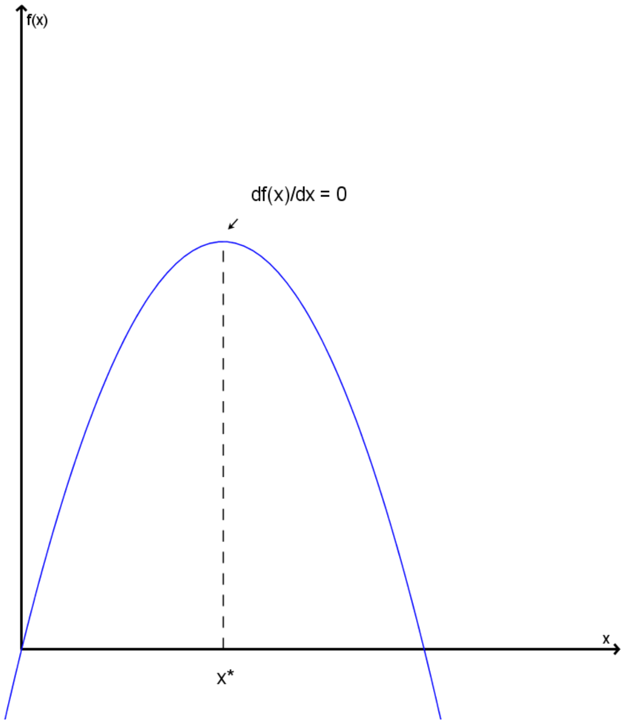
Αρχίζουμε συζήτηση για το πώς λύνονται μερικά από τα είδη των προβλημάτων που περιγράψαμε στο προηγούμενο κεφάλαιο, εξετάζοντας την απλούστερη περίπτωση, *βελτιστοποίηση χωρίς περιορισμούς* με μία μόνο μεταβλητή x ($n = 1$), όπου η διαφορίσιμη συνάρτηση $f(x)$ προς μεγιστοποίηση είναι *κοίλη*. Έτσι, η *αναγκαία και ικανή* συνθήκη για να είναι η λύση $x = x^*$ βέλτιστη (ολικό μέγιστο) είναι:

$$\frac{df}{dx} = 0, \text{ στο } x = x^*,$$

όπως απεικονίζεται στο Σχ.3.1.1. Εάν αυτή η εξίσωση μπορεί να λυθεί άμεσα για x^*, είστε έτοιμοι. Ωστόσο, αν η $f(x)$ δεν είναι ιδιαίτερα απλή συνάρτηση, ώστε η παράγωγος να είναι γραμμική ή τετραγωνική συνάρτηση, μπορεί η εξίσωση να μην λύνεται *αναλυτικά*. Αν όχι, η *διαδικασία αναζήτησης σε μία διάσταση* παρέχει έναν απλό τρόπο *αριθμητικής* επίλυσης του προβλήματος.

Διαδικασία αναζήτησης σε μία διάσταση

Όπως και οι άλλες διαδικασίες αναζήτησης μη γραμμικού προγραμματισμού, η *διαδικασία αναζήτησης σε μία διάσταση* βρίσκει μια ακολουθία *ενδιάμεσων λύσεων*, που οδηγεί σε μία βέλτιστη λύση. Σε κάθε επανάληψη, ξεκινώντας από την τρέχουσα λύση, με συστηματική έρευνα καταλήγουμε στον προσδιορισμό μιας νέας *βελτιωμένης* ενδιάμεσης λύσης.



Σχήμα 3.1.1
Το πρόβλημα προγραμματισμού μιας μεταβλητής, χωρίς περιορισμούς, όταν η συνάρτηση είναι κοίλη.

Η ιδέα πίσω από τη διαδικασία αναζήτησης σε μία διάσταση είναι πολύ διαισθητική, δηλαδή, αν η κλίση (παράγωγος) είναι θετική ή αρνητική σε μια ενδιάμεση λύση, σίγουρα δείχνει αν η αύξηση βρίσκεται προς τα δεξιά ή προς τα αριστερά, αντίστοιχα. Έτσι, αν η παράγωγος σε ένα συγκεκριμένο x είναι *θετική*, τότε το x^* πρέπει να είναι μεγαλύτερο από αυτό το x (βλέπε Σχ.3.1.1), ώστε αυτό το x να γίνεται *κάτω φράγμα* για τις ενδιάμεσες λύσεις, που θα θεωρηθούν στη συνέχεια. Αντιστρόφως, εάν η παράγωγος είναι *αρνητική*, τότε το x^* πρέπει να είναι μικρότερο από το x, ώστε το x να καθίσταται *άνω φράγμα*. Ως εκ τούτου, αφού τα δύο είδη φραγμάτων έχουν εντοπιστεί, κάθε νέα ενδιάμεση λύση, που επιλέγεται μεταξύ των τρεχόντων ορίων, παρέχει ένα νέο αυστηρότερο φράγμα του ενός τύπου, περιορίζοντας έτσι περαιτέρω την αναζήτηση. Εφ’ όσον χρησιμοποιείται ένας λογικός κανόνας, για την επιλογή κάθε ενδιάμεσης λύσης, με τον τρόπο αυτό, η προκύπτουσα *ακολουθία* των ενδιάμεσων λύσεων πρέπει να *συγκλίνει* στο x^*. Στην πράξη, αυτό σημαίνει ότι συνεχίζουμε την ακολουθία μέχρις ότου η απόσταση μεταξύ των ορίων είναι επαρκώς μικρή, ώστε η επόμενη ενδιάμεση λύση να είναι μέσα σε μια προκαθορισμένη *ανοχή σφάλματος* του x^*.

Όλη αυτή η διαδικασία συνοψίζεται στη συνέχεια, δεδομένης της παρακάτω σημειογραφίας:

x'= τρέχουσα ενδιάμεση λύση,

$\underline{x}$ = τρέχον κάτω φράγμα του x^*,

$\overline{x}$ = τρέχον ανώτατο φράγμα του x^*,

ϵ = ανοχή σφάλματος για το x^*.

Παρόλο που υπάρχουν αρκετοί λογικοί κανόνες για την επιλογή κάθε νέας ενδιάμεσης λύσης, αυτός που χρησιμοποιείται στη διαδικασία που ακολουθεί είναι **κανόνας του μέσου** (παραδοσιακά ονομάζεται *μέθοδος Bolzano*), η οποία λέει απλώς ότι επιλέγουμε τον μέσο μεταξύ των δύο τρεχόντων φραγμάτων.

Περίληψη της διαδικασίας αναζήτησης σε μία διάσταση.

Αρχικοποίηση: Επίλεξε ένα ϵ. Βρες ένα αρχικό $\underline{x}$ και $\bar{x}$, με επιθεώρηση (ή αντίστοιχα βρίσκοντας οποιαδήποτε τιμή του x, στην οποία η παράγωγος είναι θετική και, στη συνέχεια, αρνητική). Επίλεξε μια αρχική ενδιάμεση λύση

$$x' = \frac{\underline{x}+\overline{x}}{2}.$$

Επαναλήψεις:

1. Αξιολόγησε την $\frac{df(x)}{dx}$, στο $x = x'$.

2. Αν η $\frac{df(x)}{dx} \geq 0$, τότε $\underline{x} = x'$.

3. Αν η $\frac{df(x)}{dx} \leq 0$, τότε $\overline{x} = x'$.

4. Επίλεξε ένα νέο $x' = \frac{\underline{x}+\overline{x}}{2}$.

Κριτήριο τερματισμού: Αν το $\underline{x} - \overline{x} \leq 2\epsilon$, ώστε το νέο x' να είναι μικρότερο από το ϵ του x^*, τότε σταμάτα. Αλλιώς, κάνε επόμενη επανάληψη.

Θα παρουσιάσουμε τώρα αυτή τη διαδικασία με την εφαρμογή της στο ακόλουθο παράδειγμα.

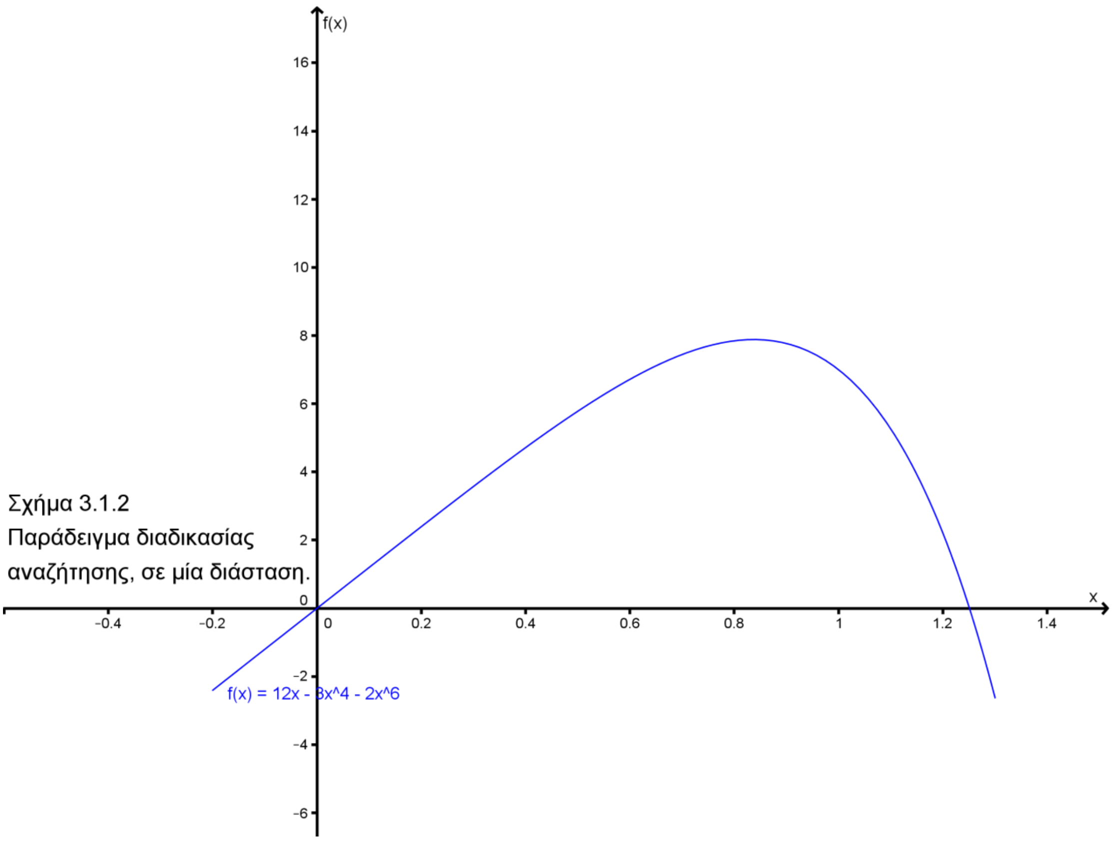
Παράδειγμα. Έστω, ότι η συνάρτηση που πρέπει να μεγιστοποιηθεί είναι η εξής:

$$f(x) = 12x - 3x^4 - 2x^6,$$

όπως έχει σχεδιαστεί στο Σχ. 3.1.2. Οι πρώτες δύο παράγωγοί της είναι:

$$\frac{df(x)}{dx} = 12(1 - x^3 - x^5),$$

$$\frac{d^2f(x)}{dx^2} = -12(3x^2 + 5x^4).$$



Σχήμα 3.1.2
Παράδειγμα διαδικασίας αναζήτησης, σε μία διάσταση.

Επανάληψη	$\frac{df(x)}{dx}$	$\underline{x}$	$\overline{x}$	**Νέο** x'	$f(x')$
0		0	2	1	7.0000
1	-12	0	1	0.5	5.7812
2	+10.12	0.5	1	0.75	7.6948
3	+4.09	0.75	1	0.875	7.8439
4	-2.19	0.75	0.875	0.8125	7.8672
5	+1.31	0.8125	0.875	0.84375	7.8829
6	-0.34	0.8125	0.84375	0.828125	7.8815
7	+0.51	0.828125	0.84375	0.8359375	7.8839
Stop					

Πίνακας 3.1.I Εφαρμογή της διαδικασίας αναζήτησης, σε μία διάσταση, στο παράδειγμα.

Επειδή, η δεύτερη παράγωγος δεν είναι θετική παντού, η *f(x)* είναι κοίλη συνάρτηση, ώστε η διαδικασία αναζήτησης σε μία διάσταση να μπορεί να εφαρμοστεί με ασφάλεια για να βρει το ολικό μέγιστο (υποθέτοντας ότι το ολικό μέγιστο υπάρχει).

Μια γρήγορη επιθεώρηση αυτής της συνάρτησης (χωρίς καν την κατασκευή γραφήματος, όπως φαίνεται στο Σχ. 3.1.2) δείχνει ότι η *f(x)* είναι θετική, για μικρές θετικές τιμές του *x*, αλλά είναι αρνητική για $x < 0$ ή $x > 2$. Συνεπώς, τα $\underline{x} = 0$ και $\overline{x} = 2$ μπορούν να χρησιμοποιηθούν ως αρχικά όρια, με τον μέσο τους, το $x' = 1$, ως αρχική ενδιάμεση λύση. Έστω, $\epsilon = 0{,}01$ η ανοχή σφάλματος για το x^* στο κριτήριο τερματισμού, ώστε τελικά $(\underline{x} - \overline{x}) \leq 0.02$ με το τελικό x' στον μέσο.

Εφαρμόζοντας διαδικασία αναζήτησης σε μία διάσταση, οδηγούμαστε στην ακολουθία αποτελεσμάτων που παρουσιάζονται στον Πίνακα 3.1.I [Ο πίνακας αυτός περιλαμβάνει τις τιμές τόσο της συνάρτησης και των παραγώγων, όπου η παράγωγος αξιολογείται ως προς την ενδιάμεση λύση, που παράγεται κατά την *τελευταία* επανάληψη. Ωστόσο, σημειώνουμε ότι ο αλγόριθμος στην πραγματικότητα δεν χρειάζεται καν να υπολογίσει την $f(x')$ και ότι υπολογίζει την παράγωγο αρκετά μεγάλου βαθμού, για να προσδιοριστεί το πρόσημό της.]

Το συμπέρασμα είναι ότι:

$$x^* \approx 0.836,$$

$$0.828125 < x^* < 0.84375.$$

3.2 Βελτιστοποίηση προβλημάτων πολλών μεταβλητών, χωρίς περιορισμούς

Ας θεωρήσουμε το πρόβλημα μεγιστοποίησης μιας κοίλης συνάρτησης $f(x)$, πολλαπλών μεταβλητών $x = (x_1, x_2,..., x_n)$, όταν δεν υπάρχουν περιορισμοί στις εφικτές τιμές. Ας υποθέσουμε, πάλι, ότι η αναγκαία και ικανή συνθήκη για την βελτιστότητα, που δίνεται από το σύστημα των εξισώσεων που λαμβάνεται με την επιβολή οι αντίστοιχες μερικές παράγωγοι να είναι ίσες με μηδέν (βλέπε Εν.2.5), δεν μπορεί να επιλυθεί αναλυτικά, έτσι ώστε να πρέπει να χρησιμοποιηθεί μια αριθμητική διαδικασία αναζήτησης. Πώς μπορεί η προηγούμενη μονοδιάστατη διαδικασία αναζήτησης να επεκταθεί και σε αυτό το πολυδιάστατο πρόβλημα;

Στην Εν.3.1, η τιμή της συνήθους παραγώγου χρησιμοποιήθηκε για να επιλέξουμε μία από τις δύο πιθανές κατευθύνσεις (αύξηση ή μείωση του x), στην οποία να μετακινηθούμε από την τρέχουσα ενδιάμεση λύση στην επόμενη. Ο στόχος ήταν να φτάσουμε σε ένα σημείο όπου τελικά αυτή η παράγωγος είναι (πρακτικά) 0. Τώρα, υπάρχουν αμέτρητες πιθανές κατευθύνσεις στις οποίες μπορούμε να μετακινηθούμε. Αυτές αντιστοιχούν στους πιθανούς *αναλογικούς ρυθμούς*, με τους οποίους μπορεί να μεταβάλλονται οι αντίστοιχες μεταβλητές. Ο στόχος είναι να φτάσουμε σε ένα σημείο όπου τελικά όλες οι μερικές παράγωγοι είναι (πρακτικά) 0. Ως εκ τούτου, η επέκταση της μονοδιάστατης διαδικασίας αναζήτησης απαιτεί να χρησιμοποιήσουμε τις τιμές των μερικών παραγώγων, για να επιλέξουμε τη συγκεκριμένη κατεύθυνση στην οποία θα κινηθούμε. Αυτή η επιλογή περιλαμβάνει τη χρήση της κλίσης της αντικειμενικής συνάρτησης, όπως περιγράφεται στη συνέχεια.

Επειδή, η αντικειμενική συνάρτηση $f(x)$ υποτίθεται ότι είναι διαφορίσιμη, έχει κλίση, που συμβολίζεται με $\nabla f(x)$, σε κάθε σημείο x. Πιο συγκεκριμένα, η **κλίση** σε ένα συγκεκριμένο σημείο $x = x'$ είναι το *διάνυσμα*, του οποίου τα στοιχεία είναι οι αντίστοιχες *μερικές παράγωγοι*, υπολογιζόμενες στο $x = x'$, έτσι ώστε:

$$\nabla f(x') = (\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n}), \text{ στο } x = x'.$$

Η σημασία της κλίσης είναι ότι η (απειροελάχιστη) μεταβολή στο *x*, που μεγιστοποιεί το ποσοστό στο οποίο η *f(x)* αυξάνει, είναι η αλλαγή που είναι ανάλογη με την $\nabla f(x)$. Για να εκφράσουμε αυτή την ιδέα γεωμετρικά, η «κατεύθυνση» της κλίσης $\nabla f(x')$ ερμηνεύεται ως η *κατεύθυνση* του προσανατολισμένου ευθύγραμμου τμήματος (βέλους), από την αρχή *(0, 0, ..., 0)* στο σημείο $(\partial f/\partial x_1, \partial f/\partial x_2, ..., \partial f/\partial x_n)$, όπου η $\partial f/\partial x_j$ εκτιμάται στο $x_j = x_j'$. Ως εκ τούτου, μπορεί να λεχθεί ότι ο ρυθμός με τον οποίο αυξάνεται η *f(x)* μεγιστοποιείται αν οι (απειροελάχιστες) μεταβολές του *x* είναι στην κατεύθυνση της κλίσης $\nabla f(x)$. Επειδή ο στόχος είναι να βρεθεί η εφικτή λύση που μεγιστοποιεί την *f(x)*, φαίνεται σκόπιμο να επιχειρήσουμε να κινηθούμε προς την κατεύθυνση της κλίσης όσο το δυνατόν περισσότερο.

Η διαδικασία αναζήτησης μέσω κλίσης

Επειδή, το συγκεκριμένο πρόβλημα δεν έχει περιορισμούς, η ερμηνεία της κλίσης υποδηλώνει ότι μια αποτελεσματική διαδικασία αναζήτησης θα πρέπει να συνεχίσει να κινείται προς την κατεύθυνση της κλίσης μέχρι να φθάσει σε μία βέλτιστη λύση x^*, όπου $\nabla f(x^*) = 0$. Ωστόσο, δεν θα ήταν πρακτικό να αλλάζουμε το *x* *συνεχώς* στην κατεύθυνση του $\nabla f(x)$, επειδή αυτή η σειρά αλλαγών θα απαιτούσε συνεχή *επανεκτίμηση* της $\partial f/\partial x_j$ και αλλαγή στην κατεύθυνση της διαδρομής. Ως εκ τούτου, μια καλύτερη προσέγγιση είναι να συνεχίσουμε να κινούμαστε σε μία *καθορισμένη* κατεύθυνση από την τρέχουσα ενδιάμεση λύση, σταματώντας μόνο όταν η *f(x)* σταματήσει να αυξάνεται. Αυτό το σημείο ακινητοποίησης θα είναι η επόμενη ενδιάμεση λύση, έτσι ώστε η κλίση, στη συνέχεια, θα πρέπει να υπολογιστεί εκ νέου για να καθορίσει την νέα κατεύθυνση στην οποία θα κινηθούμε. Με την προσέγγιση αυτή, κάθε επανάληψη περιλαμβάνει την αλλαγή της τρέχουσας ενδιάμεσης λύσης x' ως εξής:

Επανάφερε: $x' = x' + t^* \nabla f(x')$,

όπου το t^* είναι η θετική τιμή του t, που *μεγιστοποιεί* την ποσότητα

$f(x' + t\,\nabla f(x'))$, δηλαδή

$$f\left(x' + t^*\,\nabla f(x')\right) = \max_{t\geq 0} f(x' + t\,\nabla f(x')),$$

[Να σημειώσουμε ότι η ποσότητα $f(x' + t\,\nabla f(x'))$ είναι απλά η *f(x)*, όπου

$$x_j = x_j' + t\left(\frac{\partial f}{\partial x_j}\right)_{x=x'},\ \text{για}\ j = 1, 2, \ldots, n,$$

και ότι αυτές οι εκφράσεις για το x_j περιλαμβάνουν μόνο σταθερές και το *t*, έτσι ώστε η *f(x)* να είναι συνάρτηση μόνο της μεταβλητής *t*.] Οι επαναλήψεις αυτής της διαδικασίας αναζήτησης μέσω κλίσης συνεχίζονται μέχρι να είναι $\nabla f\ (x) = 0$, στα όρια μιας μικρής ανοχής ϵ. Δηλαδή, μέχρι:

$$\left|\frac{\partial f}{\partial x_j}\right| \leq \epsilon,\ \text{για}\ j = 1, 2, \ldots, n.$$

Μια αναλογία μπορεί να βοηθήσει, ώστε να αποσαφηνιστεί αυτή η διαδικασία. Ας υποθέσουμε ότι θα χρειαστεί να ανεβούμε στην κορυφή ενός λόφου. Έχουμε μυωπία, έτσι δεν μπορούμε να δούμε την κορυφή του λόφου για να περπατήσουμε άμεσα προς αυτήν την κατεύθυνση. Ωστόσο, όταν στεκόμαστε, μπορούμε να δούμε το έδαφος γύρω από τα πόδια μας αρκετά καλά, για να καθορίσουμε την κατεύθυνση προς την οποία ο λόφος έχει κλίση προς τα πάνω πιο απότομα. Είμαστε σε θέση να περπατήσουμε σε μια ευθεία γραμμή. Ενώ περπατάμε, είμαστε, επίσης, σε θέση να πούμε πότε θα σταματήσουμε την αναρρίχηση (μηδενική κλίση προς την κατεύθυνση μας). Υποθέτοντας ότι ο λόφος είναι κοίλος, μπορούμε να χρησιμοποιήσουμε την διαδικασία αναζήτησης, μέσω κλίσης, για να αναρριχηθούμε στην κορυφή αποτελεσματικά. Αυτό το πρόβλημα είναι ένα πρόβλημα δύο μεταβλητών, όπου (x_1,x_2) αντιπροσωπεύουν τις συντεταγμένες (αγνοώντας το ύψος) της τρέχουσας τοποθεσίας μας. Η συνάρτηση $f(x_1,x_2)$ δίνει το ύψος του λόφου στο σημείο (x_1,x_2). Μπορούμε να ξεκινήσουμε κάθε επανάληψη στην τρέχουσα τοποθεσία μας (τρέχουσα ενδιάμεση λύση), με τον καθορισμό της κατεύθυνσης [στο σύστημα συντεταγμένων (x_1, x_2)], στην οποία ο λόφος είναι επικλινής προς τα πάνω πιο έντονα (η κατεύθυνση της κλίσης) σε αυτό το σημείο. Μπορούμε, στη συνέχεια, να αρχίσουμε το περπάτημα σε αυτή την καθορισμένη κατεύθυνση και να συνεχίσουμε για όσο διάστημα εξακολουθούμε να αναρριχόμαστε. Μπορεί, τελικά, να σταματήσουμε σε μια νέα ενδιάμεση θέση (λύση), όταν ο λόφος γίνεται επίπεδος στην κατεύθυνσή μας, σε όποιο σημείο μπορούμε να προετοιμαστούμε, για να κάνουμε μια άλλη επανάληψη σε μια άλλη κατεύθυνση. Θα συνεχίσουμε αυτές τις επαναλήψεις, ακολουθώντας μια τεθλασμένη διαδρομή μέχρι το λόφο, μέχρι να φτάσουμε σε μία ενδιάμεση θέση, όπου η κλίση θα είναι ουσιαστικά μηδέν, σε όλες τις κατευθύνσεις. Με βάση την παραδοχή ότι ο λόφος [$f(x_1, x_2)$] είναι κοίλος, η θέση αυτή θα πρέπει να είναι ουσιαστικά στην κορυφή του λόφου.

Το πιο δύσκολο μέρος της διαδικασίας αναζήτησης μέσω κλίσης, συνήθως, είναι να βρούμε το t^*, την τιμή του t που μεγιστοποιεί την f, στην κατεύθυνση της κλίσης, σε κάθε επανάληψη. Επειδή, οι x και $\nabla f(x)$ έχουν σταθερές τιμές για τη μεγιστοποίηση και επειδή η $f(x)$ είναι κοίλη, το πρόβλημα αυτό θα πρέπει να θεωρηθεί ως ένα πρόβλημα μεγιστοποίησης μιας *κοίλης* συνάρτησης *μιας μεταβλητής* t. Ως εκ τούτου, μπορεί να λυθεί με τη διαδικασία αναζήτησης σε μία διάσταση, της Εν. 3.1 (όπου το αρχικό κάτω φράγμα για το t πρέπει να είναι μη αρνητικό, λόγω του περιορισμού $t \geq 0$). Εναλλακτικά, εάν η f είναι απλή συνάρτηση, μπορεί να είναι δυνατό να ληφθεί μία αναλυτική λύση θέτοντας την παράγωγο, ως προς το t, ίση με μηδέν και επιλύοντας.

Περίληψη της διαδικασίας αναζήτησης μέσω κλίσης.

Αρχικοποίηση: Επίλεξε το ϵ και κάθε αρχική ενδιάμεση λύση x'. Πήγαινε πρώτα στο κριτήριο τερματισμού.

Επαναλήψεις:

1. Έκφρασε την $f(x' + t\,\nabla f(x'))$ ως συνάρτηση του t, θέτοντας:

$x_j = x'_j + t(\frac{\partial f}{\partial x_j})_{x=x'}$, για $j = 1, 2, \dots, n$

και στη συνέχεια αντικαθιστώντας αυτές τις εκφράσεις στην $f(x)$.

2. Χρησιμοποίησε τη μονοδιάστατη διαδικασία αναζήτησης

(ή λογισμό), για να βρεις ένα $t = t^*$, που να μεγιστοποιεί την

$f(x' + t\,\nabla f(x'))$, όταν $t \geq 0$.

3. Επανάφερε $x' = x' + t^*\,\nabla f(x')$. Μετά, πήγαινε στο κριτήριο τερματισμού.

Κριτήριο τερματισμού: Εκτίμησε την $\nabla f(x')$ στο $x = x'$. Έλεγξε αν:

$\left|\frac{\partial f}{\partial x_j}\right| \leq \epsilon$, για όλα τα $j = 1, 2, \ldots, n$.

Αν ναι, σταμάτα, με το τρέχον x' να είναι η επιθυμητή προσέγγιση της βέλτιστης λύσης x^*. Αλλιώς, κάνε επόμενη επανάληψη.

Τώρα, ας παρουσιάσουμε αυτήν την διαδικασία.

Παράδειγμα. Θεωρούμε το ακόλουθο πρόβλημα δύο μεταβλητών:

Μεγιστοποίησε: $f(x) = 2x_1x_2 + 2x_2 - x_1^2 - 2x_2^2$,

Έτσι,

$$\frac{\partial f}{\partial x_1} = 2x_2 - 2x_1,$$

$$\frac{\partial f}{x_2} = 2x_1 + 2 - 4x_2.$$

Μπορούμε, επίσης, να ελέγξουμε αν η *f(x)* είναι κοίλη. Για να ξεκινήσει η διαδικασία αναζήτησης μέσω κλίσης, ας υποθέσουμε ότι το $x = (0,0)$ επιλέγεται ως αρχική ενδιάμεση λύση. Επειδή, οι αντίστοιχες μερικές παράγωγοι είναι *0* και *2* σε αυτό το σημείο, η κλίση είναι:

$$\nabla f(0,0) = (0,2).$$

Ως εκ τούτου, για να ξεκινήσουμε την πρώτη επανάληψη, θέτουμε:

$$x_1 = 0 + t(0) = 0,$$

$$x_2 = 0 + t(2) = 2t,$$

και στη συνέχεια, αντικαθιστούμε στην *f(x)*, για να πάρουμε:

$$\begin{aligned} f\big(x' + t\,\nabla f(x')\big) &= f(0,2t) \\ &= 2(0)(2t) + 2(2t) - 0^2 - 2(2t)^2 \\ &= 4t - 8t^2. \end{aligned}$$

Επειδή,

$$f(0,2t^*) = \max_{t\geq 0} f(0,2t) = \max_{t\geq 0}\{4t - 8t^2\}$$

και

$$\frac{d}{dt}(4t - 8t^2) = 4 - 16t = 0,$$

προκύπτει ότι:

$$t^* = \frac{1}{4},$$

οπότε:

Επανάφερε: $x' = (0,0) + \frac{1}{4}(0,2) = (0, \frac{1}{2})$.

Για αυτήν την ενδιάμεση λύση, η κλίση είναι:

$$\nabla f\left(0, \frac{1}{2}\right) = (1,0).$$

Έτσι, στη δεύτερη επανάληψη, θέτουμε:

$$x = \left(0, \frac{1}{2}\right) + t(1,0) = (t, \frac{1}{2}),$$

οπότε:

$$f\big(x' + t\,\nabla f(x')\big) = f\left(0 + t, \frac{1}{2} + 0t\right) = f\left(t, \frac{1}{2}\right)$$

$$= (2t)\left(\frac{1}{2}\right) + 2\left(\frac{1}{2}\right) - t^2 - 2(\frac{1}{2})^2$$

$$= t - t^2 + \frac{1}{2}.$$

Επειδή,

$$f\left(t^*, \frac{1}{2}\right) = \max_{t\geq 0} f(t. \frac{1}{2}) = \max_{t\geq 0}\{t - t^2 + \frac{1}{2}\}$$

και

$$\frac{d}{dt}\left(t - t^2 + \frac{1}{2}\right) = 1 - 2t = 0,$$

έπεται ότι:

$t^* = \frac{1}{2}$,

οπότε:

Επανάφερε: $x' = \left(0, \frac{1}{2}\right) + \frac{1}{2}(1,0) = (\frac{1}{2}, \frac{1}{2})$.

Ένας καλός τρόπος για την οργάνωση αυτού του έργου είναι να φτιάξουμε έναν πίνακα, όπως ο Πίνακας 3.2.Ι, που συνοψίζει τις δύο παραπάνω επαναλήψεις. Σε κάθε επανάληψη, η δεύτερη στήλη δείχνει την τρέχουσα ενδιάμεση λύση και η δεξιά στήλη δείχνει την ενδεχόμενη νέα ενδιάμεση λύση, η οποία στη συνέχεια μεταφέρεται κάτω στην δεύτερη στήλη, για την επόμενη επανάληψη. Η τέταρτη στήλη δίνει τις εκφράσεις για το x_j σαν συνάρτηση του t, που πρέπει να αντικαθίσταται στην $f(x)$, για να δώσει την πέμπτη στήλη.

Συνεχίζοντας με αυτό τον τρόπο, οι επόμενες ενδιάμεσες λύσεις θα είναι $\left(\frac{1}{2}, \frac{3}{4}\right), \left(\frac{3}{4}, \frac{3}{4}\right), \left(\frac{3}{4}, \frac{7}{8}\right), \left(\frac{7}{8}, \frac{7}{8}\right), \ldots$, όπως φαίνεται στο Σχ. 3.2.1. Επειδή, τα σημεία αυτά συγκλίνουν στο $x^* = (1,1)$, αυτή η λύση είναι η βέλτιστη λύση, όπως επαληθεύεται από το γεγονός ότι:

$\nabla f(1,1) = (0,0)$.

Επαν/ Ληψη	$\boldsymbol{x'}$	$\boldsymbol{\nabla f(x')}$	$\boldsymbol{x' + t\,\nabla f(x')}$	$\boldsymbol{f(x' + t\,\nabla f(x'))}$	$\boldsymbol{t^*}$	$\boldsymbol{x' + t^*\,\nabla f(x')}$
1	$(0,0)$	$(0,2)$	$(0,2t)$	$4t - 8t^2$	$\frac{1}{4}$	$(0, \frac{1}{2})$
2	$(0, \frac{1}{2})$	$(1,0)$	$(t, \frac{1}{2})$	$t - t^2 + \frac{1}{2}$	$\frac{1}{2}$	$(\frac{1}{2}, \frac{1}{2})$

Πίνακας 3.2.Ι Εφαρμογή της διαδικασίας αναζήτησης μέσω κλίσης στο παράδειγμα.

Ωστόσο, επειδή αυτή η συγκλίνουσα ακολουθία ενδιάμεσων λύσεων δεν φθάνει ποτέ το όριό της, η διαδικασία στην πραγματικότητα θα σταματήσει κάπου (ανάλογα το ϵ) ελαφρώς κάτω από το σημείο (1,1), ως η τελική προσέγγιση του x^*.

Όπως προτείνει το Σχ.3.2.1, η διαδικασία αναζήτησης μέσω κλίσης κινείται «ζιγκ-ζαγκ» προς τη βέλτιστη λύση και όχι σε μια ευθεία γραμμή. Έχουν αναπτυχθεί ορισμένες τροποποιήσεις της διαδικασίας, που *επιταχύνουν* την κίνηση προς τη βέλτιστη λύση, λαμβάνοντας υπόψη αυτήν την «ζιγκ-ζαγκ» συμπεριφορά.

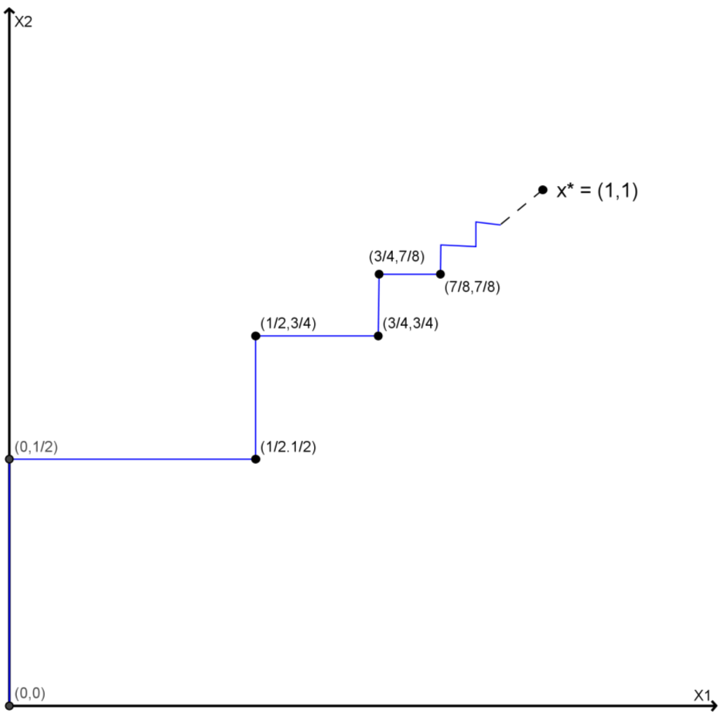
Αν η *f(x)* δεν ήταν κοίλη συνάρτηση, η διαδικασία αναζήτησης μέσω κλίσης θα συνέκλινε και πάλι σε ένα *τοπικό* μέγιστο. Η μόνη αλλαγή στην περιγραφή της διαδικασίας για την περίπτωση αυτή είναι ότι το $t*$ τώρα θα αντιστοιχούσε στο *πρώτο τοπικό μέγιστο* της $f\left(x' + t\, \nabla f(x')\right)$, καθώς το t αυξάνει από το μηδέν.

Αν, αντιθέτως, ο στόχος ήταν να *ελαχιστοποιηθεί* η *f(x)*, μία αλλαγή στη διαδικασία θα ήταν να κινηθούμε προς την *αντίθετη* κατεύθυνση της κλίσης σε κάθε επανάληψη. Με άλλα λόγια, ο κανόνας για να πάρουμε το επόμενο σημείο θα ήταν:

Επανάφερε $x' = x' - t^* \,\nabla f(x')$.

Η μόνη διαφορά είναι ότι το t^* τώρα θα ήταν η μη αρνητική τιμή του t, που *ελαχιστοποιεί* την ποσότητα $f\left(x' - t^* \,\nabla f(x')\right)$, δηλαδή:

$$f\left(x' - t^* \,\nabla f(x')\right) = \min_{t \geq 0} f(x' - t\, \nabla f(x')).$$



Σχήμα 3.2.1
Αναπαράσταση της διαδικασίας αναζήτησης μέσω βαθμίδας, όταν
f(X1,X2) = 2 X1 X2 + 2 X2 - X1^2 - 2 X2^2

Κεφάλαιο 4: Βελτιστοποίηση με περιορισμούς

4.1 Οι συνθήκες Karush-Kuhn-Tucker (KKT), για βελτιστοποίηση με περιορισμούς

Μπορούμε τώρα να επικεντρωθούμε στο ζήτημα του πώς να αναγνωρίζουμε μια *βέλτιστη λύση*, για ένα πρόβλημα μη γραμμικού προγραμματισμού (με διαφορίσιμες συναρτήσεις). Ποιες είναι οι απαραίτητες και (ίσως) επαρκείς προϋποθέσεις, τις οποίες μια τέτοια λύση πρέπει να ικανοποιεί;

Στα προηγούμενα κεφάλαια έχουμε ήδη επισημάνει τις συνθήκες για την *βελτιστοποίηση χωρίς περιορισμούς*, όπως συνοψίζεται στις δύο πρώτες σειρές του Πίνακα 4.1.I. Νωρίτερα, στην Εν. 2.5, δώσαμε επίσης αυτές τις προϋποθέσεις για την μικρή *επέκταση* της βελτιστοποίησης χωρίς περιορισμούς, όπου έχουμε *μόνο* περιορισμούς μη αρνητικότητας. Οι όροι αυτοί φαίνονται στην τρίτη σειρά του Πίνακα 4.1.I. Όπως φαίνεται στην τελευταία σειρά του πίνακα, οι συνθήκες για τη γενική περίπτωση ονομάζονται **συνθήκες Karush-Kuhn-Tucker** (ή **συνθήκες KKT**), επειδή προέκυψαν ανεξάρτητα από τον Karush, τον Kuhn και τον Tucker. Το βασικό αποτέλεσμά τους είναι ενσωματωμένο στο ακόλουθο θεώρημα:

Θεώρημα: Ας υποθέσουμε ότι οι $f(x), g_1(x), g_2(x), \dots, g_m(x)$ είναι *διαφορίσιμες* συναρτήσεις, οι οποίες πληρούν ορισμένες συνθήκες κανονικότητας. Τότε, η:

$$x^* = (x_1^*, x_2^*, \dots, x_n^*)$$

μπορεί να είναι η *βέλτιστη λύση* για το πρόβλημα μη γραμμικού προγραμματισμού, μόνο εφόσον υπάρχουν m τον αριθμό $u_1, u_2, \dots, u_m$, ώστε να πληρούνται όλες οι ακόλουθες προϋποθέσεις KKT:

$$\left.\begin{array}{l}\textbf{1. } \dfrac{\partial f}{\partial x_j} - \sum_{i=1}^{m} u_i \dfrac{\partial g_i}{\partial x_j} \leq 0 \\[2ex] \textbf{2. } x_j^* \left(\dfrac{\partial f}{\partial x_j} - \sum_{i=1}^{m} u_i \dfrac{\partial g_i}{\partial x_j}\right) = 0\end{array}\right\} \text{ στο } x = x^*, \text{ για } j = 1, 2, \dots, n.$$

$$\left.\begin{array}{l}\textbf{3. } g_i(x^*) - b_i \leq 0 \\[1ex] \textbf{4. } u_i[g_i(x^*) - b_i] = 0\end{array}\right\} \text{ για } i = 1, 2, \dots, m.$$

5. $x_j^* \geq 0,$ για $j = 1, 2, \dots, n.$

6. $u_i \geq 0,$ για $i = 1, 2, \dots, m.$

Πρόβλημα	**Αναγκαίες συνθήκες**	**Ικανές συνθήκες**
Μιας μεταβλητής, χωρίς περιορισμούς	$\frac{df}{dx} = 0$	*f(x)* κοίλη
Πολλών μεταβλητών, χωρίς περιορισμούς	$\frac{\partial f}{\partial x_j} = 0$ $(j = 1, 2, \dots, n)$	*f(**x**)* κοίλη
Με περιορισμούς μη αρνητικότητας μόνο	$\frac{\partial f}{\partial x_j} = 0$ $(j = 1, 2, \dots, n)$ (ή ≤ 0, αν $x_j = 0$)	*f(**x**)* κοίλη
Γενικά περιορισμένο πρόβλημα	Συνθήκες Karush-Kuhn-Tucker	*f(**x**)* κοίλη και $g_i(\boldsymbol{x})$ κυρτή *(i = 1, 2,…, m)*

Πίνακας 4.1.I Ικανές και αναγκαίες συνθήκες βελτιστότητας.

Να σημειώσουμε ότι οι συνθήκες 2 και 4 προϋποθέτουν ότι το γινόμενο των δύο ποσοτήτων είναι μηδέν. Ως εκ τούτου, αυτές οι συνθήκες σημαίνουν ότι τουλάχιστον μία από τις δύο ποσότητες πρέπει να είναι μηδέν. Κατά συνέπεια, η συνθήκη 4 μπορεί να συνδυαστεί με τη συνθήκη 3 και η ισοδύναμη έκφρασή τους να πάρει την εξής μορφή:

(3,4) $g_i(x^*) - b_i = 0$

(ή ≤ 0 αν $u_i = 0$), για *i = 1, 2, . . . , m.*

Όμοια, οι συνθήκες 2 και 1 μπορούν να συνδυαστούν, δίνοντας:

(1,2) $\frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} u_i \frac{\partial g_i}{\partial x_j} = 0$

(ή ≤ 0 αν $x_j^* = 0$), για $j = 1, 2, \ldots, n$.

Όταν το $m = 0$ (χωρίς συναρτήσεις περιορισμών), αυτή η άθροιση πέφτει έξω και η συνδυαστική συνθήκη (1,2) συμπυκνώνεται στη συνθήκη, που δίνεται στην τρίτη σειρά του Πίνακα 4.1.Ι. Έτσι, για $m > 0$, κάθε όρος στο άθροισμα τροποποιεί την συνθήκη για $m = 0$, ώστε να ενσωματώσει την επίδραση της αντίστοιχης συνάρτησης περιορισμού.

Στις συνθήκες 1, 2, 4 και 6, τα u_i αντιστοιχούν στις *διπλές μεταβλητές* του γραμμικού προγραμματισμού και έχουν συγκρίσιμη οικονομική ερμηνεία. Ωστόσο, τα u_i στην πραγματικότητα προέκυψαν, κατά τη μαθηματική επεξεργασία *ως πολλαπλασιαστές Lagrange*. Οι συνθήκες 3 και 5 δεν κάνουν τίποτα περισσότερο από το να εξασφαλίζουν το να είναι εφικτή η λύση. Οι υπόλοιπες συνθήκες εξαλείφουν τις περισσότερες από τις εφικτές λύσεις, ως πιθανές υποψήφιες, βέλτιστες λύσεις.

Ωστόσο, σημειώνουμε ότι το γεγονός ικανοποιούνται οι συνθήκες δεν εγγυάται ότι η λύση είναι η βέλτιστη. Όπως συνοψίζεται στη δεξιά στήλη του Πίνακα 4.1.Ι, απαιτούνται ορισμένες επιπλέον υποθέσεις *κυρτότητας*, για την απόκτηση αυτής της εγγύησης. Αυτές οι υποθέσεις διατυπώνονται στην ακόλουθη επέκταση του θεωρήματος.

Πόρισμα. Ας υποθέσουμε ότι η $f(x)$ είναι *κοίλη* συνάρτηση και οι $g_1(x)$, $g_2(x), \ldots, g_m(x)$ είναι *κυρτές* συναρτήσεις (δηλαδή, αυτό το πρόβλημα είναι ένα πρόβλημα κυρτού προγραμματισμού), όπου όλες οι συναρτήσεις να πληρούν τις συνθήκες κανονικότητας. Τότε η $x^* = (x_1^*, x_2^*, \ldots, x_n^*)$ αποτελεί τη *βέλτιστη λύση*, αν και μόνο αν πληρούνται όλες οι προϋποθέσεις του θεωρήματος.

Παράδειγμα. Για να δείξουμε τη διαμόρφωση και την εφαρμογή των συνθηκών ΚΚΤ, θεωρούμε το ακόλουθο πρόβλημα μη γραμμικού προγραμματισμού, με δύο μεταβλητές:

Μεγιστοποίησε: $f(x) = \ln(x_1 + 1) + x_2$, υπό τους περιορισμούς: $2x_1 + x_2 \leq 3$

και

$$x_1 \geq 0, x_2 \geq 0,$$

όπου το ln συμβολίζει το φυσικό λογάριθμο. Έτσι, $m = 1$ (μία συνάρτηση περιορισμού) και $g_1(x) = 2x_1 + x_2$, οπότε η $g_1(x)$ είναι κυρτή. Επιπλέον, μπορεί εύκολα να επαληθευτεί ότι η $f(x)$ είναι κοίλη. Ως εκ τούτου, το Πόρισμα εφαρμόζεται, έτσι ώστε οποιαδήποτε λύση, που πληροί τις προϋποθέσεις KKT, σίγουρα να είναι μια βέλτιστη λύση. Εφαρμόζοντας τους τύπους, που δίνονται στο θεώρημα, προκύπτουν οι ακόλουθες προϋποθέσεις KKT για αυτό το παράδειγμα:

1(j = 1). $\frac{1}{x_1+1} - 2u_1 \leq 0.$

2(j = 1). $x_1\left(\frac{1}{x_1+1} - 2u_1\right) = 0.$

1(j = 2). $1 - u_1 \leq 0.$

2(j = 2). $x_2(1 - u_1) = 0.$

3. $2x_1 + x_2 - 3 \leq 0.$

4. $u_1(2x_1 + x_2 - 3) = 0.$

5. $x_1 \geq 0, x_2 \geq 0.$

6. $u_1 \geq 0.$

Τα βήματα επίλυσης των συνθηκών KTT, για το συγκεκριμένο πρόβλημα, περιγράφονται παρακάτω:

1. $u_1 \geq 0$, από τη συνθήκη 1(*j = 2*)

$x_1 \geq 0$, από τη συνθήκη 5.

2. Ως εκ τούτου, $\frac{1}{x_1+1} - 2u_1 < 0.$

3. Ως εκ τούτου, $x_1 = 0$, από τη συνθήκη 2(*j = 1*).

4. $u_1 \neq 0$ συνεπάγεται ότι $2x_1 + x_2 - 3 = 0$, από τη συνθήκη 4.

5. Τα βήματα 3 και 4 συνεπάγονται ότι $x_2 = 3$.

6. $x_2 \neq 0$ συνεπάγεται ότι $u_1 = 1$, από τη συνθήκη 2 ($j = 2$).

7. Δεν παραβιάζονται οι συνθήκες, όταν $x_1 = 0$, $x_2 = 3$, $u_1 = 1$.

Ως εκ τούτου, υπάρχει ένας αριθμός $u_1 = 1$, έτσι ώστε $x_1 = 0$, $x_2 = 3$ και η $u_1 = 1$ πληροί όλες τις προϋποθέσεις. Κατά συνέπεια, η $x^* = (0,3)$ αποτελεί τη βέλτιστη λύση, για αυτό το πρόβλημα.

Η συγκεκριμένη εξέλιξη των βημάτων, που απαιτούνται για την επίλυση των συνθηκών ΚΚΤ, θα διαφέρει από το ένα πρόβλημα στο επόμενο. Όταν δεν είναι εμφανής η λογική που ακολουθείται, μερικές φορές είναι χρήσιμο να εξεταστούν χωριστά οι διάφορες περιπτώσεις, όπου κάθε x_j και u_i τίθεται να είναι είτε ίσο ή μεγαλύτερο από 0 και στη συνέχεια, δοκιμάζοντας κάθε περίπτωση μέχρι κάποια να οδηγήσει σε λύση. Στο παράδειγμα, υπάρχουν οκτώ τέτοιες περιπτώσεις, που αντιστοιχούν στους οκτώ συνδυασμούς $x_1 = 0$ σε σχέση με $x_1 > 0$, $x_2 = 0$ έναντι $x_2 > 0$ και $u_1 = 0$ σε σχέση με $u_1 > 0$. Κάθε περίπτωση οδηγεί σε μία απλούστερη κατάσταση και ανάλυση των συνθηκών. Παρουσιάζουμε παρακάτω την περίπτωση, όπου $x_1 = 0$, $x_2 = 0$, $u_1 = 0$.

Συνθήκες ΚΤΤ, για την περίπτωση $x_1 = 0$, $x_2 = 0$, $u_1 = 0$

1(*j* = *1*). $\frac{1}{0+1} \leq 0$. Άτοπο.

1(*j* = *2*). $1 - 0 \leq 0$. Άτοπο.

3. $0 + 0 \leq 3$.

(Όλες οι άλλες συνθήκες είναι περιττές.)

Όπως αναφέρεται παρακάτω, οι άλλες τρεις περιπτώσεις, όπου $u_1 = 0$ δίνουν επίσης άμεσα άτοπο, με παρόμοιο τρόπο, οπότε δεν υπάρχει διαθέσιμη λύση.

Για $x_1 = 0$, $x_2 > 0$, $u_1 = 0$ έχουμε άτοπο στις συνθήκες:

1($j = 1$), 1($j = 2$) και 2($j = 2$).

Για $x_1 > 0$, $x_2 = 0$, $u_1 = 0$ έχουμε άτοπο στις συνθήκες:

$1(j = 1)$, $2(j = 1)$ και $1(j = 2)$.

Για $x_1 > 0, x_2 > 0, u_1 = 0$ έχουμε άτοπο στις συνθήκες:

$1(j = 1)$, $2(j = 1)$, $1(j = 2)$ και $2(j = 2)$.

Η περίπτωση $x_1 > 0, x_2 > 0, u_1 > 0$ επιτρέπει τη διαγραφή των μη μηδενικών πολλαπλασιαστών από τις συνθήκες $2(j = 1)$, $2(j = 2)$, και 4, η οποία στη συνέχεια επιτρέπει τη διαγραφή των συνθηκών $1(j = 1)$, $1(j = 2)$, και 3, επειδή περιττεύει, όπως συνοψίζεται στη συνέχεια.

Συνθήκες ΚΤΤ, για την περίπτωση $x_1 > 0, x_2 > 0, u_1 > 0$

1(*j* = *1*). $\frac{1}{x_1+1} - 2u_1 = 0.$

2(*j* = *2*). $1 - u_1 = 0.$

4. $2x_1 + x_2 - 3 = 0.$

(Οι υπόλοιπες συνθήκες είναι περιττές.)

Ως εκ τούτου, $u_1 = 0$, οπότε $x_1 = -\frac{1}{2}$, το οποίο είναι άτοπο, αφού $x_1 > 0$.

Ας θεωρήσουμε, τώρα, την περίπτωση $x_1 = 0, x_2 > 0, u_1 > 0$.

Συνθήκες ΚΤΤ, για την περίπτωση $x_1 = 0, x_2 > 0, u_1 > 0$

1(*j* = *1*). $\frac{1}{0+1} - 2u_1 = 0.$

2(*j* = *2*). $1 - u_1 = 0.$

4. $0 + x_2 - 3 = 0.$

(Οι υπόλοιπες συνθήκες είναι περιττές.)

Άρα, $x_1 = 0$, $x_2 = 3$, $u_1 = 1$. Εφόσον, έχουμε βρει μία λύση, γνωρίζουμε ότι δεν χρειάζεται να εξετάσουμε άλλες περιπτώσεις.

Για προβλήματα πιο περίπλοκα από αυτό το παράδειγμα, μπορεί να είναι δύσκολο, αν όχι αδύνατο, να αντλήσουμε μια βέλτιστη λύση *άμεσα*, από τις συνθήκες ΚΚΤ. Παρ' όλα αυτά, αυτές οι συνθήκες εξακολουθούν να παρέχουν πολύτιμες ενδείξεις ως προς την ταυτότητα της βέλτιστης λύσης και μας επιτρέπουν, επίσης, να ελέγξουμε κατά πόσον η προτεινόμενη λύση μπορεί να είναι η βέλτιστη.

Υπάρχουν, επίσης, πολλές πολύτιμες *έμμεσες* εφαρμογές των προϋποθέσεων ΚΚΤ. Μία από αυτές τις εφαρμογές εμφανίζεται στη *θεωρία δυϊκότητας*, για μη γραμμικό προγραμματισμό. Ειδικότερα, για κάθε δεδομένο πρόβλημα μεγιστοποίησης με περιορισμούς (το ονομάσουμε *πρωταρχικό πρόβλημα*), οι συνθήκες ΚΚΤ μπορεί να χρησιμοποιηθούν για να καθορίσουμε ένα στενά συνδεδεμένο δυϊκό πρόβλημα, που είναι πρόβλημα ελαχιστοποίησης με περιορισμούς. Οι μεταβλητές στο δυϊκό πρόβλημα αποτελούνται τόσο από τους πολλαπλασιαστές Lagrange u_i ($i = 1, 2, . . . , m$) και τις πρωταρχικές μεταβλητές x_j ($j = 1, 2, . . . , n$). Στην ειδική περίπτωση, όπου το βασικό πρόβλημα είναι ένα πρόβλημα γραμμικού προγραμματισμού, οι μεταβλητές x_j βγαίνουν εκτός του δυϊκού προβλήματος και έχουμε το απλό δυϊκό πρόβλημα γραμμικού προγραμματισμού. Όταν το πρωταρχικό πρόβλημα είναι πρόβλημα κυρτού προγραμματισμού, είναι δυνατόν να αποδειχθούν σχέσεις μεταξύ πρωταρχικού προβλήματος και το δυϊκού προβλήματος, που είναι παρόμοιες με εκείνες για το γραμμικό προγραμματισμό. Για παράδειγμα, η *ισχυρή ιδιότητα δυϊκότητας* , η οποία ορίζει ότι οι βέλτιστες τιμές της αντικειμενικής συνάρτησης των δύο προβλημάτων είναι ίσες, ισχύει και εδώ. Επιπλέον, οι τιμές των u_i μεταβλητών, σε μια βέλτιστη λύση για το δυϊκό πρόβλημα, μπορεί και πάλι να ερμηνευθούν ως *σκιώδεις τιμές*, δηλαδή, δίνουν το ρυθμό με τον οποίο η βέλτιστη τιμή της αντικειμενικής συνάρτησης για το πρωταρχικό πρόβλημα θα μπορούσε να αυξηθεί αυξάνοντας ελαφρά τη δεξιά πλευρά του αντίστοιχου περιορισμού. Επειδή, η θεωρία δυϊκότητας, για μη γραμμικό προγραμματισμό είναι ένα σχετικά προχωρημένο θέμα, ο ενδιαφερόμενος αναγνώστης παραπέμπεται αλλού για περαιτέρω πληροφόρηση.

Θα δούμε μια άλλη, έμμεση εφαρμογή των συνθηκών ΚΚΤ, στην επόμενη ενότητα.

4.2 Τετραγωνικός Προγραμματισμός

Όπως αναφέρεται στην Εν.2.5, το πρόβλημα τετραγωνικού προγραμματισμού διαφέρει από ένα πρόβλημα γραμμικού προγραμματισμού μόνο στο γεγονός ότι η αντικειμενική συνάρτηση περιλαμβάνει, επίσης, τους όρους x_j^2 και $x_i x_j$ $(i \neq j)$. Έτσι, αν χρησιμοποιήσουμε μητρωική γραφή, το πρόβλημα είναι να βρούμε ένα $\boldsymbol{x}$, έτσι ώστε να ισχύει:

Μεγιστοποίησε: $f(\boldsymbol{x}) = \boldsymbol{cx} - \frac{1}{2}\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{Qx}$,

υπό τους περιορισμούς:

$\boldsymbol{Ax} \leq \boldsymbol{b}$ και $\boldsymbol{x} \geq \boldsymbol{0}$,

όπου $\boldsymbol{c}$ είναι ένα διάνυσμα-γραμμή, $\boldsymbol{x}$ και $\boldsymbol{b}$ είναι διανύσματα-στήλες, $\boldsymbol{Q}$ και $\boldsymbol{A}$ είναι πίνακες, και ο εκθέτης T υποδηλώνει τον ανάστροφο. Τα στοιχεία q_{ij} (στοιχεία του $\boldsymbol{Q}$) είναι δεδομένες σταθερές, έτσι ώστε $q_{ij} = q_{ji}$ (που είναι ο λόγος ύπαρξης του παράγοντα $\frac{1}{2}$ στην αντικειμενική συνάρτηση). Εκτελώντας τους αναφερόμενους πολλαπλασιασμούς διανύσματος-πίνακα, η αντικειμενική συνάρτηση εκφράζεται ως συνάρτηση των q_{ij}, των c_j (στοιχείων του $\boldsymbol{c}$), καθώς και των μεταβλητών, ως εξής:

$$f(\boldsymbol{x}) = \boldsymbol{cx} - \frac{1}{2}\boldsymbol{x}^{\boldsymbol{T}}\boldsymbol{Qx} = \sum_{j=1}^{n} c_j x_j - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij} x_i x_j.$$

Για κάθε όρο όπου $i = j$, σε αυτό το διπλό άθροισμα, $x_i x_j = x_j^2$, έτσι $-\frac{1}{2}q_{jj}$ είναι ο συντελεστής του x_j^2. Όταν $i \neq j$, τότε $-\frac{1}{2}\left(q_{ij}x_i x_j + q_{ji}x_j x_i\right) = -q_{ij}x_i x_j$, έτσι $-q_{ij}$ είναι ο συνολικός συντελεστής του γινομένου των x_i και x_j.

Για να παρουσιάσουμε τη χρήση των συμβόλων, θεωρούμε το ακόλουθο παράδειγμα προβλήματος τετραγωνικού προγραμματισμού.

Μεγιστοποίησε: $f(x_1, x_2) = 15x_1 + 30x_2 + 4x_1 x_2 - 2x_1^2 - 4x_2^2$,

υπό τους περιορισμούς

$$x_1 + 2x_2 \leq 30$$

και

$$x_1 \geq 0, x_2 \geq 0.$$

Σε αυτήν την περίπτωση,

$$\boldsymbol{c} = [15 \quad 30], \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \boldsymbol{Q} = \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix},$$

$$\boldsymbol{A} = [1 \quad 2], \boldsymbol{b} = [30].$$

Σημειώνουμε ότι:

$$\begin{aligned} \boldsymbol{x^T Q x} &= [x_1 \quad x_2] \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= [(4x_1 - 4x_2) \quad (-4x_1 + 8x_2)] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= 4x_1^2 - 4x_2x_1 - 4x_1x_2 + 8x_2^2 \\ &= q_{11}x_1^2 + q_{21}x_2x_1 + q_{12}x_1x_2 + q_{22}x_2^2. \end{aligned}$$

Πολλαπλασιάζοντας την παράσταση με $-\frac{1}{2}$, παίρνουμε:

$$-\frac{1}{2}\boldsymbol{x^T Q x} = -2x_1^2 + 4x_1x_2 - 4x_2^2.$$

το οποίο είναι το μη γραμμικό τμήμα της αντικειμενικής συνάρτησης, για αυτό το παράδειγμα. Εφόσον, $q_{11} = 4$ και $q_{22} = 8$, το παράδειγμα δείχνει ότι $-\frac{1}{2}q_{jj}$ είναι ο συντελεστής του x_j^2, στην αντικειμενική συνάρτηση. Το γεγονός ότι $q_{12} = q_{21} = -4$ δείχνει ότι και τα δύο $-q_{ij}$ και $-q_{ji}$ δίνουν το συνολικό συντελεστή του γινομένου των x_i και x_j.

Έχουν αναπτυχθεί αρκετοί αλγόριθμοι, για την ειδική περίπτωση του προβλήματος τετραγωνικού προγραμματισμού, όπου η αντικειμενική συνάρτηση είναι κοίλη συνάρτηση. (Ένας τρόπος για να βεβαιωθούμε ότι η αντικειμενική συνάρτηση είναι κοίλη είναι να εξετασθεί η ισοδύναμη κατάσταση, κατά την οποία:

$$\boldsymbol{x^T Q x \geq 0}$$

για όλα τα $\boldsymbol{x}$, δηλαδή, ότι ο $\boldsymbol{Q}$ είναι *θετικά ημιορισμένος* πίνακας.) Θα περιγράψουμε έναν από αυτούς τους αλγορίθμους, *την τροποποιημένη μέθοδο Simplex*, που είναι αρκετά δημοφιλής, διότι απαιτεί τη χρήση μόνο της μεθόδου *Simplex,* με μια μικρή τροποποίηση. Το κλειδί αυτής της προσέγγισης είναι να κατασκευάσει τις συνθήκες ΚΚΤ της προηγούμενης ενότητας και στη συνέχεια, να εκφράσει αυτές τις συνθήκες σε μια βολική μορφή, που μοιάζει με γραμμικό προγραμματισμό. Ως εκ τούτου, πριν από την περιγραφή του αλγορίθμου, θα αναπτύξουμε αυτή τη βολική μορφή.

Οι συνθήκες ΚΚΤ για Τετραγωνικό Προγραμματισμό

Για διατήρηση της συνάφειας, ας εξετάσουμε πρώτα το παραπάνω παράδειγμα. Ξεκινώντας με τη μορφή, που δίνεται στην προηγούμενη ενότητα, οι συνθήκες της ΚΚΤ είναι οι ακόλουθες:

1(*j* = *1*). $15 + 4x_2 - 4x_1 - u_1 \leq 0.$

2(*j* = *1*). $x_1(15 + 4x_2 - 4x_1 - u_1) = 0.$

1(*j* = *2*). $30 + 4x_1 - 8x_2 - 2u_1 \leq 0.$

2(*j* = *2*). $x_2(30 + 4x_1 - 8x_2 - 2u_1) = 0.$

3. $x_1 + 2x_2 - 30 \leq 0.$

4. $u_1(x_1 + 2x_2 - 30) = 0.$

5. $x_1 \geq 0, x_2 \geq 0.$

6. $u_1 \geq 0.$

Για να ξεκινήσουμε την εκ νέου έκφραση αυτών των συνθηκών σε μια πιο βολική μορφή, μεταβάλλουμε τις σταθερές στις συνθήκες 1(*j* = *1*), 1(*j* = *2*) και 3 στην δεξιά πλευρά και στη συνέχεια εισαγάγουμε μη αρνητικές *χαλαρές μεταβλητές* (συμβολίζονται με y_1, y_2, και v_1, αντίστοιχα), για τη μετατροπή αυτών των ανισοτήτων σε εξισώσεις.

1(*j*=*1*). $-4x_1 + 4x_2 - u_1 + y_1 = -15$

1(*j*=*2*). $4x_1 - 8x_2 - 2u_1 + y_2 = -30$

3. $x_1 + 2x_2 \qquad\qquad + v_1 = \quad 30$

Σημειώνουμε ότι η συνθήκη 2(*j = 2*) μπορεί να εκφραστεί εκ νέου, αν απλώς απαιτήσουμε $x_1 = 0$ είτε $y_1 = 0$, δηλαδή:

2(*j = 1*). $x_1 y_1 = 0.$

Όμοια, οι συνθήκες 2(*j = 2*) και 4 μπορούν να αντικατασταθούν από:

2(*j = 2*). $x_2 y_2 = 0,$

4. $u_1 v_1 = 0.$

Για καθένα από αυτά τα τρία ζεύγη- (x_1,y_1), (x_2,y_2), (u_1,v_1) -οι δύο μεταβλητές ονομάζονται **συμπληρωματικές μεταβλητές**, επειδή μόνο μία από τις δύο μεταβλητές μπορεί να είναι διαφορετική από το μηδέν. Αυτές οι νέες μορφές συνθηκών 2(*j = 1*), 2(*j = 2*), και 4 μπορούν να συνδυαστούν σε έναν περιορισμό,

$$x_1 y_1 + x_2 y_2 + u_1 v_1 = 0,$$

που ονομάζεται ο **περιορισμός συμπληρωματικότητας**.

Πολλαπλασιάζοντας τις εξισώσεις για τις συνθήκες 1(*j = 1*) και 1(*j = 2*) με -1, αποκτάμε μη αρνητικό δεξί μέλος και να έχουμε την επιθυμητή κατάλληλη μορφή, για το σύνολο των προϋποθέσεων, ως εξής:

$$\begin{aligned} 4x_1 - 4x_2 + u_1 - y_1 \qquad &= 15 \\ -4x_1 + 8x_2 + 2u_1 \quad - y_2 \qquad &= 30 \\ x_1 + 2x_2 \qquad\qquad + v_1 &= 30 \end{aligned}$$

$$x_1 \geq 0, x_2 \geq 0, u_1 \geq 0, y_1 \geq 0, y_2 \geq 0, v_1 \geq 0$$

$$x_1 y_1 + x_2 y_2 + u_1 v_1 = 0$$

Αυτή η μορφή είναι ιδιαίτερα βολική, επειδή, εκτός από τον περιορισμό συμπληρωματικότητας, αυτές οι συνθήκες είναι *περιορισμοί γραμμικού προγραμματισμού*.

Για οποιοδήποτε πρόβλημα τετραγωνικού προγραμματισμού, οι συνθήκες KKT μπορεί να απλουστευθούν σε μια τέτοια βολική μορφή, που να περιέχει μόνο περιορισμούς γραμμικού προγραμματισμού, συν ένα συμπληρωματικότητα περιορισμό. Σε μητρωική γραφή και πάλι, η γενική μορφή είναι η εξής:

$$\boldsymbol{Qx} + \boldsymbol{A^T u} - \boldsymbol{y} = \boldsymbol{c^T},$$

$$\boldsymbol{Ax} + \boldsymbol{v} = \boldsymbol{b},$$

$$\boldsymbol{x} \geq \boldsymbol{0}, \boldsymbol{u} \geq \boldsymbol{0}, \boldsymbol{y} \geq \boldsymbol{0}, \boldsymbol{v} \geq \boldsymbol{0},$$

$$\boldsymbol{x^T y} + \boldsymbol{u^T v} = \boldsymbol{0},$$

όπου τα στοιχεία του διανύσματος-στήλης $\boldsymbol{u}$ είναι τα u_i του προηγούμενου τμήματος και τα στοιχεία των διανυσμάτων-στηλών $\boldsymbol{y}$ και $\boldsymbol{v}$ είναι χαλαρές μεταβλητές.

Επειδή, η αντικειμενική συνάρτηση του αρχικού προβλήματος υποτίθεται ότι είναι κοίλη και επειδή οι συναρτήσεις περιορισμών είναι γραμμικές και επομένως, κυρτές, το πόρισμα του θεωρήματος της Εν. 4.1 ισχύει. Έτσι, το $\boldsymbol{x}$ είναι η *βέλτιστη λύση*, αν και μόνο αν υπάρχουν τιμές των $\boldsymbol{y}$, $\boldsymbol{u}$ και $\boldsymbol{v}$, έτσι ώστε και τα τέσσερα διανύσματα μαζί να ικανοποιούν όλες αυτές τις συνθήκες. Το αρχικό πρόβλημα, με αυτόν τον τρόπο, απλοποιείται στο ισοδύναμο πρόβλημα της εύρεσης μιας *εφικτή λύση*, για τους *περιορισμούς* αυτούς.

Είναι ενδιαφέρον να σημειωθεί ότι αυτό το ισοδύναμο πρόβλημα είναι ένα παράδειγμα του *γραμμικού προβλήματος συμπληρωματικότητας*, που εισήχθη στην Εν.2.5 και ότι ο περιορισμός-κλειδί, για το γραμμικό πρόβλημα της συμπληρωματικότητας, είναι ο *περιορισμός συμπληρωματικότητάς* του.

Η τροποποιημένη μέθοδος Simplex

Η *τροποποιημένη μέθοδος Simplex* εκμεταλλεύεται το βασικό γεγονός ότι, αν εξαιρέσουμε τον περιορισμό συμπληρωματικότητας, οι συνθήκες ΚΚΤ στη βολική μορφή, που λαμβάνεται παραπάνω, δεν είναι τίποτα περισσότερο από ό,τι οι περιορισμοί γραμμικού προγραμματισμού. Επιπλέον, ο περιορισμός συμπληρωματικότητας απλά σημαίνει ότι δεν επιτρέπεται *και οι δύο* συμπληρωματικές μεταβλητές του κάθε ζεύγους να είναι (μη εκφυλισμένες) βασικές μεταβλητές (οι μόνες μεταβλητές > 0), όταν εξετάζονται οι (μη εκφυλισμένες) βασικές εφικτές (Β.Ε.) λύσεις (μη αρνητικές). Ως εκ τούτου, το πρόβλημα απλοποιείται σε ένα πρόβλημα εξεύρεσης μιας αρχικής Β.Ε. λύσης, σε οποιοδήποτε πρόβλημα γραμμικού προγραμματισμού, που έχει αυτούς τους περιορισμούς, που υπόκειται στον πρόσθετο περιορισμό, σχετικά με την ταυτότητα των βασικών μεταβλητών. (Αυτή η αρχική Β.Ε. λύση μπορεί να είναι η μόνη εφικτή λύση σε αυτήν την περίπτωση.)

Η εύρεση μιας τέτοιας Β.Ε. αρχικής λύσης είναι σχετικά απλή. Στην απλή περίπτωση όπου $\boldsymbol{c^T} \leq \boldsymbol{0}$ (απίθανο) και $\boldsymbol{b} \geq \boldsymbol{0}$, οι αρχικές βασικές μεταβλητές είναι τα στοιχεία των $\boldsymbol{y}$ και $\boldsymbol{v}$ (πολλαπλασιάζουμε την πρώτη ομάδα εξισώσεων με -1), έτσι ώστε η επιθυμητή λύση να είναι $\boldsymbol{x} = \boldsymbol{0}$, $\boldsymbol{u} = \boldsymbol{0}$, $\boldsymbol{y} = -\boldsymbol{c^T}$, $\boldsymbol{v} = \boldsymbol{b}$. Διαφορετικά, θα πρέπει να αναθεωρήσουμε το πρόβλημα εισάγοντας μία *τεχνητή μεταβλητή*, σε κάθε μία από τις εξισώσεις όπου $c_j > 0$ (προσθέτουμε τη μεταβλητή στα αριστερά) ή $b_i < 0$ (αφαιρούμε τη μεταβλητή στα αριστερά και στη συνέχεια πολλαπλασιάστε με -1), προκειμένου να χρησιμοποιήσουμε αυτές τις τεχνητές μεταβλητές (τις ονομάζουμε z_1, z_2, και ούτω καθεξής) ως αρχικές βασικές μεταβλητές για το αναθεωρημένο πρόβλημα. (Σημειώνουμε ότι αυτή η επιλογή των αρχικών βασικών μεταβλητών ικανοποιεί τον περιορισμό συμπληρωματικότητας, γιατί ως μη βασικές μεταβλητές $\boldsymbol{x} = \boldsymbol{0}$ και $\boldsymbol{u} = \boldsymbol{0}$, αυτόματα).

Στη συνέχεια, βρίσκουμε Β.Ε. λύση για το πραγματικό πρόβλημα, δηλαδή, λύση που να εφαρμόζει τη μέθοδο Simplex (με μία τροποποίηση) στο ακόλουθο πρόβλημα γραμμικού προγραμματισμού:

Ελαχιστοποίησε: $Z = \sum_j z_j$,

υπό τους περιορισμούς του γραμμικού προγραμματισμού, που λαμβάνονται από τις συνθήκες ΚΚΤ, αλλά με τη χρήση τεχνητών μεταβλητών.

Μία τροποποίηση της μεθόδου Simplex είναι η ακόλουθη αλλαγή στη διαδικασία επιλογής εισαγωγής μιας βασικής μεταβλητής.

Κανόνας περιορισμένης εισόδου: Όταν επιλέγουμε εισαγωγή βασικής μεταβλητής, αποκλείουμε από την εξέταση τυχόν μη βασικές μεταβλητές, των οποίων η *συμπληρωματική μεταβλητή* είναι ήδη βασική μεταβλητή. Η επιλογή πρέπει να γίνει από τις άλλες μη βασικές μεταβλητές, σύμφωνα με το σύνηθες κριτήριο για τη μέθοδο Simplex.

Ο κανόνας αυτός ικανοποιεί τον περιορισμό της συμπληρωματικότητας, καθ 'όλη τη διάρκεια του αλγορίθμου. Όταν μια βέλτιστη λύση:

$$\boldsymbol{x}^*, \boldsymbol{u}^*, \boldsymbol{y}^*, \boldsymbol{v}^*, z_1 = 0, \ldots, z_n = 0$$

λαμβάνεται για το πρόβλημα, η $\boldsymbol{x}^*$ είναι η επιθυμητή βέλτιστη λύση, για το αρχικό πρόβλημα τετραγωνικού προγραμματισμού.

Παράδειγμα. Θα δούμε τώρα την προσέγγιση αυτή στο παράδειγμα, που δίνεται στην αρχή της ενότητας. Όπως μπορεί να επαληθευτεί, η $f(x_1,x_2)$ είναι *αυστηρά κοίλη* συνάρτηση, δηλαδή:

$$Q = \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix}$$

είναι θετικά ορισμένη, ώστε να μπορεί να εφαρμοστεί ο αλγόριθμος.

Το σημείο εκκίνησης, για την επίλυση αυτού του παραδείγματος, είναι οι συνθήκες ΚΚΤ, στην βολική μορφή, που είδαμε νωρίτερα στην ενότητα. Αφού εισαχθούν οι αναγκαίες τεχνητές μεταβλητές, το πρόβλημα γραμμικού προγραμματισμού, που πρέπει να επιλυθεί με την τροποποιημένη μέθοδο Simplex είναι το εξής:

Ελαχιστοποίησε: $Z = z_1 + z_2$,

υπό τους περιορισμούς:

$$\begin{aligned} 4x_1 - 4x_2 + u_1 - y_1 \quad\quad + z_1 \quad &= 15 \\ -4x_1 + 8x_2 + 2u_1 \quad - y_2 \quad\quad + z_2 &= 30 \\ x_1 + 2x_2 \quad\quad - v_1 \quad\quad &= 30 \end{aligned}$$

και

$$x_1 \geq 0, x_2 \geq 0, u_1 \geq 0, y_1 \geq 0, y_2 \geq 0, v_1 \geq 0,$$

$$z_1 \geq 0, z_2 \geq 0.$$

Ο επιπλέον συμπληρωματικός περιορισμός είναι:

$$x_1 y_1 + x_2 y_2 + u_1 v_1 = 0,$$

δεν περιλαμβάνεται απαραίτητα, επειδή ο αλγόριθμος εφαρμόζει αυτόματα αυτόν τον περιορισμό, λόγω του κανόνα της *περιορισμένης εισόδου*. Ειδικότερα, για καθένα από τα τρία ζεύγη των συμπληρωματικών μεταβλητών (x_1,y_1), (x_2,y_2), (u_1,v_1), όποτε μία από τις δύο μεταβλητές είναι ήδη βασική, η άλλη μεταβλητή *αποκλείεται* να εισαχθεί σαν βασική μεταβλητή. Να θυμόμαστε ότι μόνο οι βασικές μεταβλητές είναι *μη μηδενικές*. Επειδή, η αρχική σειρά βασικών μεταβλητών, για το πρόβλημα γραμμικού προγραμματισμού -z_1, z_2, v_1- δίνει μια αρχική Β.Ε. λύση, που ικανοποιεί τον περιορισμό της συμπληρωματικότητας, δεν υπάρχει κανένας τρόπος ώστε ο περιορισμός αυτός να παραβιάζεται, από κάθε μεταγενέστερη Β.Ε. λύση.

Ο Πίνακας 4.2.Ι δείχνει τα αποτελέσματα της εφαρμογής της τροποποιημένης μεθόδου Simplex, στο πρόβλημα αυτό. Το πρώτο ταμπλό Simplex εμφανίζει το αρχικό σύστημα των εξισώσεων, μετά τη μετατροπή από την ελαχιστοποίηση Z, στη μεγιστοποίηση $-Z$ και εξαλείφοντας αλγεβρικά τις αρχικές βασικές μεταβλητές από την Εξ. (0). Οι τρεις επαναλήψεις προχωρούν, ακριβώς όπως και για την απλή μέθοδο Simplex, *εκτός* από την κατάργηση ορισμένων στοιχείων για την είσοδο βασικών μεταβλητών, λόγω του κανόνα της περιορισμένης εισόδου. Στο πρώτο ταμπλό, το u_1 αποβάλλεται, διότι η συμπληρωματική μεταβλητή (v_1), είναι ήδη βασική μεταβλητή (αλλά το x_2 θα είχε επιλεγεί ούτως ή άλλως, διότι $-4 < -3$). Στο δεύτερο ταμπλό, τόσο το u_1 όσο και το y_2 εξαλείφονται (επειδή v_1 και x_2 είναι βασικές μεταβλητές), οπότε, το x_1 αυτομάτως επιλέγεται ως το μόνο με αρνητικό συντελεστή στη σειρά 0 (λαμβάνοντας υπόψη ότι η τακτική της *απλής* μεθόδου Simplex θα επέτρεπε την επιλογή x_1 ή u_1, επειδή συνδέονται με το γεγονός ότι έχουν το μεγαλύτερο αρνητικό συντελεστή). Στο τρίτο ταμπλό, τα y_1 και y_2 εξαλείφονται (επειδή τα x_1 και x_2 είναι βασικές μεταβλητές). Ωστόσο, το u_1 *δεν* έχει εξαλειφθεί, γιατί το v_1 δεν είναι πλέον βασική μεταβλητή, οπότε, το u_1 επιλέγεται ως βασική μεταβλητή, που εισέρχεται με τον συνήθη τρόπο.

Επαν/ληψη	Βασ. Μεταβ	Εξ.	Z	x_1	x_2	u_1	y_1	y_2	v_1	z_1	z_2	Δεξί μέλος
0	Z	(0)	-1	0	-4	-3	1	1	0	0	0	-45
	z_1	(1)	0	4	-4	1	-1	0	0	1	0	15
	z_2	(2)	0	-4	8	2	0	-1	0	0	1	30
	v_1	(3)	0	1	2	0	0	0	1	0	0	30
1	Z	(0)	-1	-2	0	-2	1	$\frac{1}{2}$	0	0	$\frac{1}{2}$	-30
	z_1	(1)	0	2	0	2	-1	$-\frac{1}{2}$	0	1	$\frac{1}{2}$	30
	x_2	(2)	0	$-\frac{1}{2}$	1	$\frac{1}{4}$	0	$-\frac{1}{8}$	0	0	$\frac{1}{8}$	$3\frac{3}{4}$
	v_1	(3)	0	2	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	1	0	$-\frac{1}{4}$	$22\frac{1}{2}$
2	Z	(0)	-1	0	0	$-\frac{5}{2}$	1	$\frac{3}{4}$	1	0	$\frac{1}{4}$	$-7\frac{1}{2}$
	z_1	(1)	0	0	0	$\frac{5}{2}$	-1	$-\frac{3}{4}$	-1	1	$\frac{3}{4}$	$7\frac{1}{2}$
	x_2	(2)	0	0	1	$\frac{1}{8}$	0	$-\frac{1}{16}$	$\frac{1}{4}$	0	$\frac{1}{16}$	$9\frac{3}{8}$
	x_1	(3)	0	1	0	$-\frac{1}{4}$	0	$\frac{1}{8}$	$\frac{1}{2}$	0	$-\frac{1}{8}$	$11\frac{1}{4}$
3	Z	(0)	-1	0	0	0	0	0	0	1	1	0
	u_1	(1)	0	0	0	1	$-\frac{2}{5}$	$-\frac{3}{10}$	$-\frac{2}{5}$	$\frac{2}{5}$	$\frac{3}{10}$	3
	x_2	(2)	0	0	1	0	$\frac{1}{20}$	$-\frac{1}{40}$	$\frac{3}{10}$	$-\frac{}{2}$	$\frac{1}{40}$	9
	x_1	(3)	0	1	0	0	$-\frac{1}{10}$	$\frac{1}{20}$	$\frac{2}{5}$	$\frac{1}{10}$	$-\frac{1}{20}$	12

Πίνακας 4.2.I Εφαρμογή της τροποποιημένης μεθόδου Simplex στο παράδειγμα τετραγωνικού προγραμματισμού.

Η προκύπτουσα βέλτιστη λύση είναι $x_1 = 12$, $x_2 = 9$, $u_1 = 3$, με το υπόλοιπο των μεταβλητών να ισούται με μηδέν. (Επιβεβαιώνουμε ότι η λύση αυτή είναι η βέλτιστη, δείχνοντας ότι οι $x_1 = 12$, $x_2 = 9$, $u_1 = 3$ πληρούν τις προϋποθέσεις ΚΚΤ, για το αρχικό πρόβλημα, όταν είναι γραμμένες, με τη μορφή που αναφέρονται στην Εν. 4.1). Ως εκ τούτου, η βέλτιστη λύση, για το πρόβλημα τετραγωνικού προγραμματισμού (η οποία περιλαμβάνει μόνο τις μεταβλητές x_1 και x_2) είναι η $(x_1, x_2) = (12, 9)$.

Ορισμένες επιλογές Λογισμικού

Ο οδηγός εκπαίδευσης για την Ε.Ε. περιλαμβάνει μία διαδραστική ρουτίνα, για την τροποποιημένη μέθοδο Simplex, για να μας βοηθήσει να μάθουμε αυτόν τον αλγόριθμο αποτελεσματικά. Επιπλέον, με Excel, LINGO, LINDO, και MPL / CPLEX μπορούμε να λύσουμε προβλήματα τετραγωνικού προγραμματισμού.

Η διαδικασία επίλυσης, με τη χρήση Excel είναι σχεδόν η ίδια όπως και με τη χρήση γραμμικού προγραμματισμού. Μία κρίσιμη διαφορά είναι ότι η εξίσωση που γράφεται στο κελί που περιέχει την αξία της αντικειμενικής συνάρτησης πρέπει να είναι τετραγωνική εξίσωση. Για το δούμε, εξετάζουμε πάλι το παράδειγμα, που εισαγάγαμε στην αρχή της ενότητας, το οποίο έχει την παρακάτω αντικειμενική συνάρτηση:

$$f(x_1, x_2) = 15x_1 + 30x_2 + 4x_1x_2 - 2x_1^2 - 4x_2^2.$$

Ας υποθέσουμε ότι οι τιμές των x_1 και x_2 είναι στα κελιά B4 και C4 του λογιστικού φύλλου Excel, και ότι η τιμή της αντικειμενικής συνάρτησης είναι στο κελί F4. Στη συνέχεια, η εξίσωση για το κελί F4 πρέπει να είναι:

$$\mathrm{F4} = 15 * \mathrm{B4} + 30 * \mathrm{C4} + 4 * \mathrm{B4} * \mathrm{C4} - 2 * (\mathrm{B4\hat{}2}) - 4 * (\mathrm{C4\hat{}2}),$$

όπου το σύμβολο ^2 δείχνει τον εκθέτη 2. Πριν από την επίλυση του μοντέλου, πρέπει να κάνουμε κλικ στο κουμπί Επιλογή και να βεβαιωθούμε ότι η επιλογή *Υπόθεση Γραμμικού Μοντέλου* δεν είναι επιλεγμένη (δεδομένου ότι αυτό δεν είναι μοντέλο *γραμμικού* προγραμματισμού).

Όταν χρησιμοποιούμε MPL/CPLEX, θα πρέπει να επιλέγουμε την επιλογή *Τετραγωνικά Μοντέλα* από το παράθυρο διαλόγου MPL Γλώσσας και τη *μέθοδο Φραγμού*, από το παράθυρο διαλόγου CPLEX Simplex. Διαφορετικά, η διαδικασία είναι η ίδια όπως και με γραμμικό προγραμματισμό, εκτός από το ότι η έκφραση για την αντικειμενική συνάρτηση είναι τετραγωνική συνάρτηση. Έτσι, για παράδειγμα, η αντικειμενική συνάρτηση θα εκφράζεται ως:

$$15x1 + 30x2 + 4x1 * x2 - 2(x1\hat{}2) - 4(x2\hat{}2).$$

Δε χρειάζεται να γίνει κάτι παραπάνω, κατά την κλήση CPLEX, δεδομένου ότι θα αναγνωρίσει αυτόματα το μοντέλο ως πρόβλημα τετραγωνικού προγραμματισμού.

Η συγκεκριμένη αντικειμενική συνάρτηση θα εκφραστεί με τον ίδιο τρόπο, για ένα μοντέλο LINGO. Το LINGO θα καλέσει αυτόματα το μη γραμμικό *solver* (λύτη) της, για να λύσει το μοντέλο. Όταν, αντιθέτως, χρησιμοποιούμε LINDO, η διαδικασία είναι κάπως πιο περίπλοκη, δεδομένου ότι απαιτεί τη μετατροπή του μοντέλου σε μία ισοδύναμη γραμμική μορφή, με τη χρήση συνθηκών KKT.

Μερικά από αυτά τα πακέτα λογισμικού μπορούν, επίσης, να εφαρμοστούν σε πιο πολύπλοκα είδη προβλημάτων μη γραμμικού προγραμματισμού από τα προβλήματα τετραγωνικού προγραμματισμού. Αν το πακέτο CPLEX δεν μπορεί να λύσει ένα πρόβλημα, η επαγγελματική έκδοση του MPL δεν υποστηρίζει κάποιον άλλο solver που να μπορεί. Η έκδοση του MPL για μαθητές στο CD-ROM περιλαμβάνει έναν τέτοιο solver, που καλείται CONOPT (ένα προϊόν της ARKOI Consulting), που θα πρέπει να χρησιμοποιηθεί αντί του CPLEX, αφού επιλέξουμε *Μη γραμμικά μοντέλα*, για την εισαγωγή *Προεπιλεγμένου Τύπου Μοντέλου*, στο παράθυρο διαλόγου MPL Γλώσσας. Τόσο το Excel όσο και το LINGO περιλαμβάνουν ευέλικτους solvers μη γραμμικών προβλημάτων. Ωστόσο, πρέπει να γνωρίζουν ότι αυτοί οι solvers εγγυώνται την εύρεση βέλτιστης λύσης, για περίπλοκα προβλήματα, ιδιαίτερα για προβλήματα μη κυρτού προγραμματισμού.

4.3 Διαχωρίσιμος Προγραμματισμός

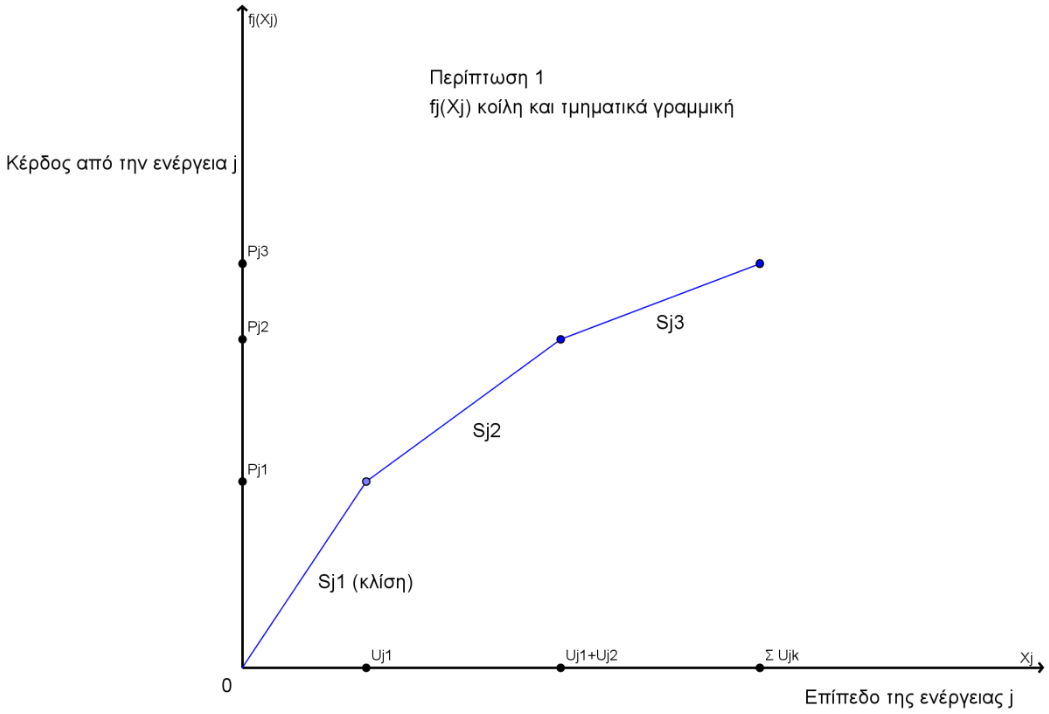
Στην προηγούμενη ενότητα είδαμε πώς μια κατηγορία προβλημάτων μη γραμμικού προγραμματισμού μπορεί να λυθεί με μια επέκταση της μεθόδου Simplex. Εμείς τώρα εξετάσουμε μια άλλη κατηγορία, που ονομάζεται *διαχωρίσιμος προγραμματισμός*, που πραγματικά μπορεί να λυθεί από την ίδια τη μέθοδο Simplex, διότι κάθε τέτοιο πρόβλημα μπορεί να προσεγγίζεται όσο είναι επιθυμητό από ένα πρόβλημα γραμμικού προγραμματισμού, με ένα μεγαλύτερο αριθμό μεταβλητών.

Όπως αναφέρεται στην Εν.2.5, σε προβλήματα διαχωρίσιμου προγραμματισμού θεωρούμε ότι η αντικειμενική συνάρτηση $f(x)$ είναι κοίλη, ότι καθεμία από τις συναρτήσεις περιορισμών $g_i(x)$ είναι κυρτή, και ότι όλες αυτές οι συναρτήσεις είναι διαχωρίσιμες (συναρτήσεις, όπου κάθε όρος περιλαμβάνει μία μόνο μεταβλητή). Ωστόσο, για να απλοποιηθεί η συζήτηση, θα επικεντρωθούμε στην ειδική περίπτωση, όπου οι κυρτές και διαχωρίσιμες $g_i(x)$ είναι γραμμικές συναρτήσεις, ακριβώς όπως και για γραμμικό προγραμματισμό. Έτσι, μόνο η αντικειμενική συνάρτηση απαιτεί ειδική μεταχείριση.

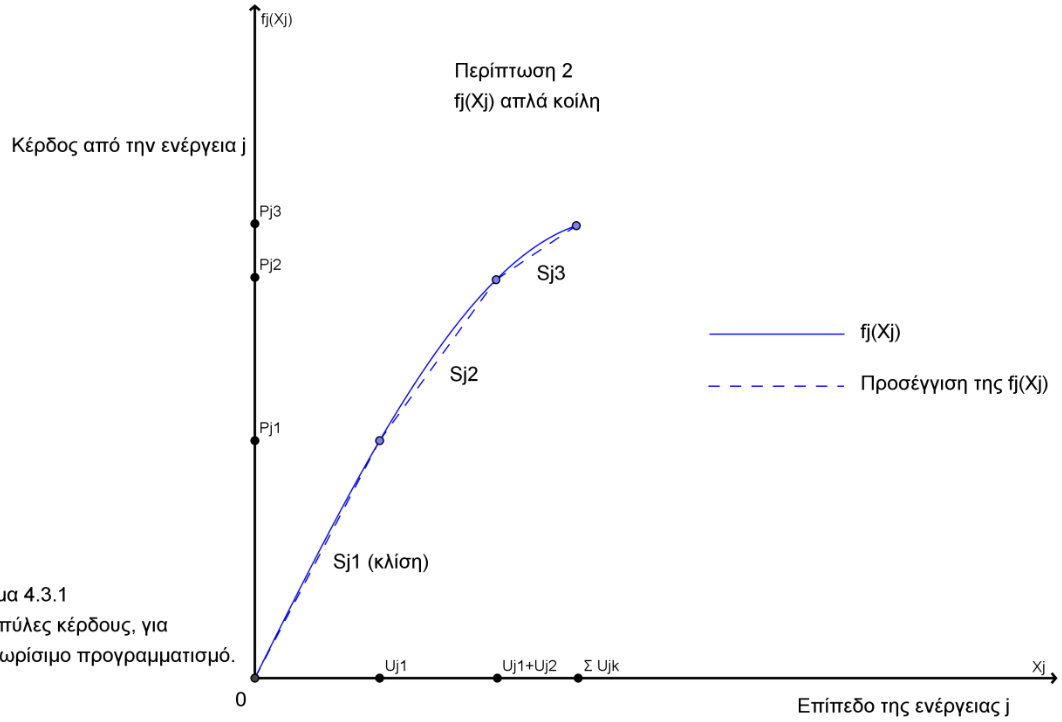
Σύμφωνα με τις προηγούμενες υποθέσεις, η αντικειμενική συνάρτηση μπορεί να εκφραστεί ως άθροισμα των κοίλων συναρτήσεων των επιμέρους μεταβλητών ως:

$$f(x) = \sum_{j=1}^{n} f_j(x_j),$$

έτσι ώστε κάθε $f_j(x_j)$ να έχει σχήμα όπως αυτό που φαίνεται στο Σχ.4.3.1 (δύο περιπτώσεις), πάνω από το εφικτό εύρος των τιμών των x_j. Επειδή, η $f(x)$ αντιπροσωπεύει το μέτρο της απόδοσης (για παράδειγμα, το κέρδος) για όλες τις δραστηριότητες από κοινού, η $f_j(x_j)$ αναπαριστά τη *συνεισφορά κέρδους*, από τη δραστηριότητα j, όταν πραγματοποιείται σε επίπεδο x_j. Το γεγονός ότι η $f(x)$ είναι διαχωρίσιμη σημαίνει απλά ότι ισχύει η προσθετικότητα, δηλαδή, δεν υπάρχουν αλληλεπιδράσεις μεταξύ των δραστηριοτήτων (χωρίς όρους γινομένου), που επηρεάζουν το συνολικό κέρδος, πέρα από την ανεξάρτητη συνεισφορά τους. Η παραδοχή ότι κάθε $f_j(x_j)$ είναι κοίλη, λέει ότι η *οριακή κερδοφορία* (κλίση της καμπύλης κέρδους) είτε παραμένει ίδια ή μειώνεται (ποτέ δεν αυξάνει), καθώς το x_j αυξάνεται.



|← x_{j1} →|←--- x_{j2} ----→|←--- x_{j3} ----→|



Σχήμα 4.3.1
Καμπύλες κέρδους, για
διαχωρίσιμο προγραμματισμό.

|← x_{j1} →|← x_{j2} →|< x_{j3} >|

Κοίλες καμπύλες κέρδους συναντάμε αρκετά συχνά. Για παράδειγμα, μπορεί να είναι δυνατό να πουλήσουμε ένα περιορισμένο ποσό κάποιου προϊόντος σε μια ορισμένη τιμή, ένα επιπλέον ποσό σε χαμηλότερη τιμή, και ίσως τελικά ένα επιπλέον ποσό σε ακόμα χαμηλότερη τιμή. Ομοίως, μπορεί να χρειαστεί να αγοράσουμε πρώτες ύλες από πηγές αυξανόμενης ακρίβειας. Σε μια άλλη κοινή κατάσταση, πρέπει να χρησιμοποιηθεί μια πιο δαπανηρή διαδικασία παραγωγής (π.χ. υπερωρίες, αντί για εργασία τακτικής απασχόλησης), για την αύξηση του ρυθμού παραγωγής, πέρα από ένα ορισμένο σημείο.

Αυτά τα είδη καταστάσεων μπορεί να οδηγήσουν σε οποιονδήποτε τύπο καμπύλης κέρδους, όπως φαίνεται στο Σχ.4.3.1. Στην περίπτωση 1, η κλίση μειώνεται μόνο σε ορισμένα *σημεία διακοπής*, έτσι ώστε η $f_j(x_j)$ να είναι *κατά τμήματα γραμμική συνάρτηση* (μια ακολουθία συνδεόμενων ευθύγραμμων τμημάτων). Στην περίπτωση 2, η κλίση μπορεί να μειωθεί καθώς αυξάνεται συνεχώς το x_j, ώστε η $f_j(x_j)$ να είναι γενικά κοίλη συνάρτηση. Οποιαδήποτε τέτοια συνάρτηση μπορεί να προσεγγίζεται, κατά το επιθυμητό, από μια τμηματικά γραμμική συνάρτηση, και αυτό το είδος της προσέγγισης χρησιμοποιείται ανάλογα με τις ανάγκες των προβλημάτων διαχωρίσιμου προγραμματισμού. (Το Σχ.4.3.1 δείχνει μια προσεγγιστική συνάρτηση, που αποτελείται από μόλις τρία ευθύγραμμα τμήματα, αλλά η προσέγγιση μπορεί να γίνει ακόμα καλύτερα, απλά με την εισαγωγή πρόσθετων σημείων διακοπής.) Αυτή η προσέγγιση είναι πολύ βολική, γιατί μια τμηματικά γραμμική συνάρτηση μιας μεταβλητής μπορεί να ξαναγραφεί ως μια *γραμμική συνάρτηση* πολλών μεταβλητών, με έναν ειδικό περιορισμό των τιμών αυτών των μεταβλητών, όπως περιγράφεται στη συνέχεια.

Αναδιατύπωση ως Πρόβλημα Γραμμικού Προγραμματισμού

Το «κλειδί», για την αναδιατύπωση μιας τμηματικά γραμμικής συνάρτησης ως γραμμικής, είναι να χρησιμοποιήσουμε μια διαφορετική μεταβλητή, για κάθε τμήμα γραμμής. Για να το δούμε, ας εξετάσουμε την τμηματικά γραμμική συνάρτηση $f_j(x_j)$, που φαίνεται στο Σχ.4.3.1, στην περίπτωση 1 (ή η προσεγγιστική τμηματικά γραμμική συνάρτηση, για την περίπτωση 2), η οποία έχει τρία ευθύγραμμα τμήματα πάνω από το εφικτό εύρος τιμών των x_j. Εισάγουμε τις τρεις νέες μεταβλητές x_{j1}, x_{j2}, και x_{j3}και θέτουμε:

$$x_j = x_{j1} + x_{j2} + x_{j3},$$

όπου

$$0 \leq x_{j1} \leq u_{j1}, 0 \leq x_{j2} \leq u_{j2}, 0 \leq x_{j3} \leq u_{j3}.$$

Στη συνέχεια, χρησιμοποιούμε τις κλίσεις s_{j1}, s_{j2} και s_{j3}, για να ξαναγράψουμε την $f_j(x_j)$, ως εξής:

$$f_j\left(x_j\right) = s_{j1}x_{j1} + s_{j2}x_{j2} + s_{j3}x_{j3},$$

με τον *ειδικό περιορισμό* ότι:

$$x_{j2} = 0 \text{ όποτε } x_{j1} < u_{j1},$$

$$x_{j3} = 0 \text{ όποτε } x_{j2} < u_{j2}.$$

Για να δούμε γιατί είναι απαραίτητος αυτός ο ειδικός περιορισμός, υποθέτουμε ότι $x_j = 1$, όπου $u_{jk} > 1$ $(k = 1, 2, 3)$, έτσι ώστε $f_j(1) = s_{j1}$. Σημειώνουμε ότι:

$$x_{j1} + x_{j2} + x_{j3} = 1$$

επιτρέπει:

$$x_{j1} = 1, x_{j2} = 0, x_{j3} = 0 => f_j(1) = s_{j1},$$

$$x_{j1} = 0, x_{j2} = 1, x_{j3} = 0 => f_j(1) = s_{j2},$$

$$x_{j1} = 0, x_{j2} = 0, x_{j3} = 1 => f_j(1) = s_{j3},$$

και ούτω καθεξής, όπου:

$$s_{j1} > s_{j2} > s_{j3}.$$

Ωστόσο, ο ειδικός περιορισμός επιτρέπει μόνο την πρώτη περίπτωση, η οποία είναι η μόνη που παρέχει τη σωστή τιμή για το $f_j(1)$.

Δυστυχώς, ο ειδικός περιορισμός δεν ταιριάζει στην απαιτούμενη μορφή περιορισμών γραμμικού προγραμματισμού, έτσι μερικές τμηματικά γραμμικές συναρτήσεις δεν μπορούν να ξαναγραφτούν σε γραμμική μορφή προγραμματισμού. Ωστόσο, οι $f_j(x_j)$ θεωρούνται κοίλες, ώστε $s_{j1} > s_{j2} > \cdots$, έτσι ώστε ένας αλγόριθμος μεγιστοποίησης της $f(x)$ να δίνει *αυτόματα* την υψηλότερη προτεραιότητα στη χρήση της x_{j1}, όταν (σε ισχύ) αυξάνουμε το x_j από το μηδέν, την επόμενη υψηλότερη προτεραιότητα στη χρήση x_{j2}, και ούτω καθεξής, χωρίς ακόμη να συμπεριλάβουμε σαφώς τον ειδικό περιορισμό στο μοντέλο. Η παρατήρηση αυτή οδηγεί στην ακόλουθη βασική ιδιότητα.

Βασική Ιδιότητα του Διαχωρίσιμου προγραμματισμού. Όταν οι $f(x)$ και $g_i(x)$ ικανοποιούν τις υποθέσεις του διαχωρίσιμου προγραμματισμού, και όταν οι κατά τμήματα γραμμικές συναρτήσεις, που προκύπτουν ξαναγραφτούν, ως γραμμικές συναρτήσεις, διαγράφοντας τον *ειδικό περιορισμό*, παίρνουμε ένα *μοντέλο γραμμικού προγραμματισμού*, του οποίου η βέλτιστη λύση ικανοποιεί αυτόματα τον ειδικό περιορισμό.

Θα επεξεργαστούμε περαιτέρω τη λογική πίσω από αυτή τη βασική ιδιότητα αργότερα σε αυτήν την ενότητα, στο πλαίσιο ενός συγκεκριμένου παραδείγματος.

Για να γράψουμε το πλήρες μοντέλο γραμμικού προγραμματισμού της παραπάνω σημείωσης, ας θεωρήσουμε n_j τον αριθμό των γραμμικών τμημάτων της $f_j(x_j)$ (ή την τμηματικά γραμμική προσέγγισή της), έτσι ώστε:

$$x_j = \sum_{k=1}^{n_j} x_{jk}$$

να αντικατασταθεί σε όλο το αρχικό μοντέλο και:

$$f_j(x_j) = \sum_{k=1}^{n_j} s_{jk} x_{jk}$$

να αντικατασταθεί στην αντικειμενική συνάρτηση για $j = 1, 2, \ldots, n$. Το προκύπτον μοντέλο είναι το εξής:

Μεγιστοποίησε: $Z = \sum_{j=1}^{n} (\sum_{k=1}^{n_j} s_{jk} x_{jk})$,

υπό τους περιορισμούς:

$\sum_{j=1}^{n} a_{ij}(\sum_{k=1}^{n_j} x_{jk}) \leq b_i$, για $i = 1, 2, \ldots, m$

$x_{jk} \leq u_{jk}$, για $k = 1, 2, \ldots, n_j, j = 1, 2, \ldots, n$

και

$x_{jk} \geq 0$, για $k = 1, 2, \ldots, n_j, j = 1, 2, \ldots, n.$

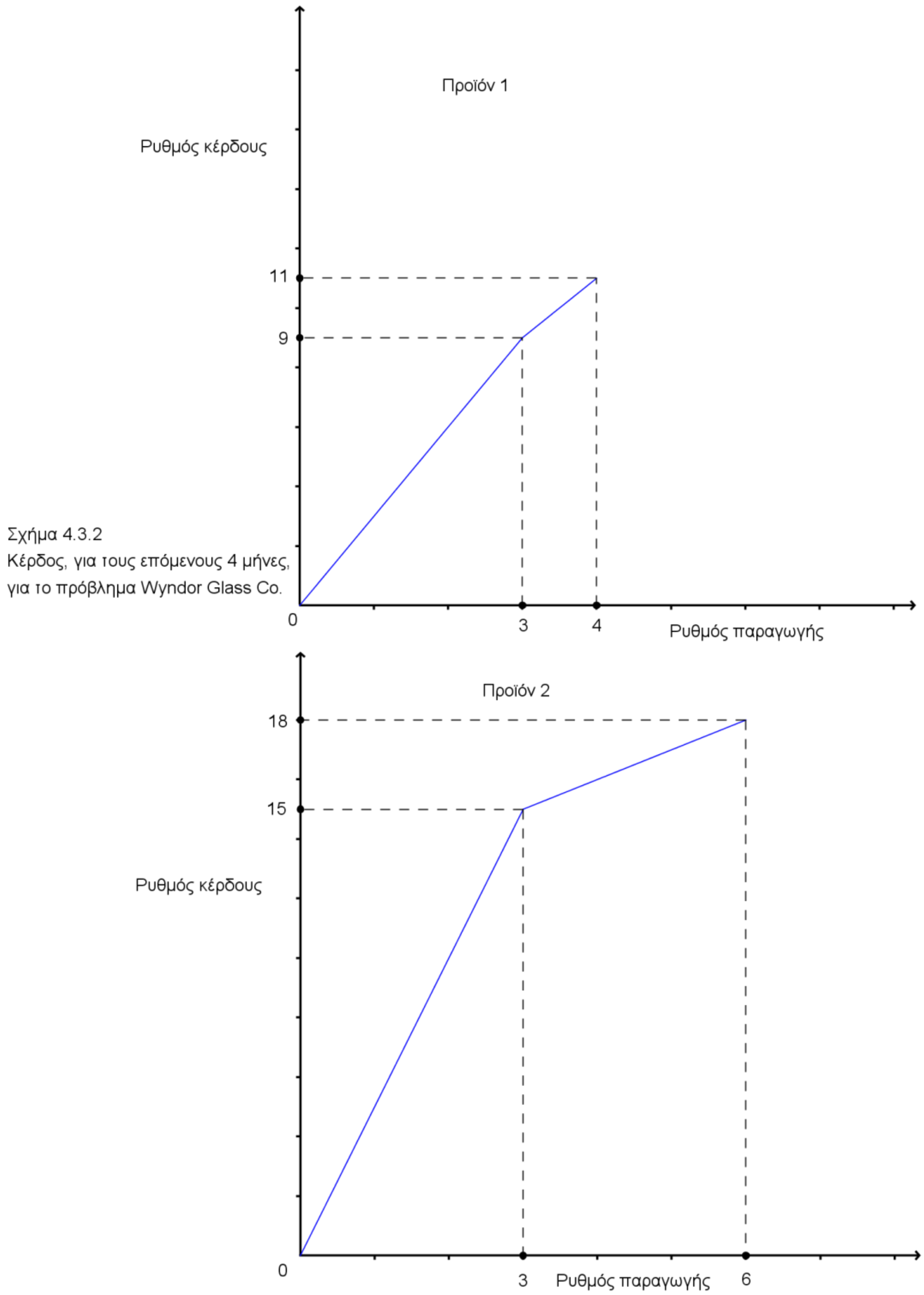
(Οι περιορισμοί $\sum_{k=1}^{n_j} x_{jk} \geq 0$ διαγράφονται, εφόσον εξασφαλίζονται από τους περιορισμούς $x_{jk} \geq 0$.) Εάν κάποια αρχική μεταβλητή x_j δεν έχει άνω φράγμα, τότε $u_{jn_j} = \infty$, οπότε ο περιορισμός που περιλαμβάνει αυτήν την ποσότητα θα διαγραφεί.

Ένας αποτελεσματικός τρόπος επίλυσης αυτού του μοντέλου είναι να χρησιμοποιήσουμε μία βελτιωμένη εκδοχή της μεθόδου Simplex, για την αντιμετώπιση περιορισμών για το άνω φράγμα. Μετά την απόκτηση μιας βέλτιστης λύσης, για το συγκεκριμένο μοντέλο, τότε θα υπολογίσουμε την ποσότητα:

$x_j = \sum_{k=1}^{n_j} x_{jk},$

για $j = 1, 2, \ldots, n$, προκειμένου να εντοπίσουμε μια βέλτιστη λύση, για το αρχικό πρόγραμμα διαχωρίσιμου προγραμματισμού (ή για την τμηματικά γραμμική προσέγγισή του).

Παράδειγμα. Το πρόβλημα Wyndor Glass Co. (βλέπε Εν.2.4) έχει λάβει ειδική εντολή για κάποια χειροποίητα αγαθά, που πρέπει να παραχθούν στα Εργοστάσια 1 και 2, μέσα στους επόμενους 4 μήνες. Για να εκπληρωθεί αυτός ο σκοπός απαιτεί το δανεισμό ορισμένων κατηγοριών εργαζομένων, από το προσωπικό για τα συνήθη προϊόντα, έτσι ώστε οι υπόλοιποι εργαζόμενοι θα πρέπει να εργάζονται υπερωρίες, για να αξιοποιήσουν την πλήρη παραγωγική ικανότητα των μηχανημάτων και εξοπλισμού του σταθμού, για τα κανονικά προϊόντα. Ειδικότερα, για τα δύο νέα προϊόντα που συζητήθηκαν στην Εν.2.4, θα πρέπει να γίνουν υπερωρίες, για να χρησιμοποιηθεί το τελευταίο 25% της παραγωγικής ικανότητας του Εργοστασίου 1, για το προϊόν 1 και για το τελευταίο 50% της διαθέσιμης χωρητικότητας Εργοστασίου 2, για το προϊόν 2. Το επιπλέον κόστος της χρήσης υπερωριακής απασχόλησης θα μειώσει το κέρδος, για κάθε μονάδα που συμμετέχει από 3$ έως 2$, για το προϊόν 1 και από 5$ έως 1$, για το προϊόν 2, δίνοντας τις καμπύλες κέρδους του Σχ.4.3.2, οι οποίες ταιριάζουν με τη μορφή της περίπτωσης 1, του Σχ.4.3.1.



Σχήμα 4.3.2
Κέρδος, για τους επόμενους 4 μήνες, για το πρόβλημα Wyndor Glass Co.

Η διοίκηση έχει αποφασίσει να προχωρήσει στη χρησιμοποίηση υπερωριακής εργασίας, αντί να προσλάβει επιπλέον εργαζόμενους, κατά τη διάρκεια αυτής της προσωρινής κατάστασης. Ωστόσο, επιμένει ότι το προσωπικό, για κάθε προϊόν πρέπει να αξιοποιηθεί πλήρως στον κανονικό χρόνο, πριν χρησιμοποιηθεί οποιαδήποτε υπερωρία. Επιπλέον, θεωρεί ότι οι τρέχοντες συντελεστές παραγωγής ($x_1 = 2$ για το προϊόν 1 και $x_2 = 6$ για το προϊόν 2) θα πρέπει να αλλάξουν προσωρινά, εάν αυτό θα βελτιώσει τη συνολική κερδοφορία. Ως εκ τούτου, ανέθεσε στην ομάδα Ε.Ε. να ελέγξει ξανά τα προϊόντα 1 και 2, για να καθορίσει την πιο αποδοτική σύνθεση προϊόντων, κατά τη διάρκεια των επόμενων 4 μηνών.

Διατύπωση. Ας θυμηθούμε το μοντέλο γραμμικού προγραμματισμού, για το αρχικό πρόβλημα Wyndor Glass Co. της Εν.2.4, που είναι το εξής:

Μεγιστοποίησε: $Z = 3x_1 + 5x_2$,

υπό τους περιορισμούς:

$$\begin{aligned} x_1 \quad\quad &\leq 4 \\ 2x_2 &\leq 12 \\ 3x_1 + 2x_2 &\leq 18 \end{aligned}$$

και

$$x_1 \geq 0, x_2 \geq 0.$$

Πρέπει τώρα να τροποποιήσουμε αυτό το μοντέλο για να ταιριάζει στη νέα κατάσταση, που περιγράφηκε παραπάνω. Για το σκοπό αυτό, έστω ότι ο ρυθμός παραγωγής του προϊόντος 1 είναι: $x_1 = x_{1R} + x_{1O}$, όπου x_{1R} είναι ο ρυθμός παραγωγής που επιτυγχάνεται στην κανονική περίοδο και x_{1O} είναι ο σταδιακά αυξανόμενος ρυθμός παραγωγής, κατά τη διάρκεια υπερωριών. Ορίζουμε: $x_2 = x_{2R} + x_{2O}$ κατά τον ίδιο τρόπο για το προϊόν 2. Έτσι, στο συμβολισμό του γενικού μοντέλου γραμμικού προγραμματισμού, για τον διαχωρίσιμο προγραμματισμό, που παρουσιάστηκε προηγουμένως, είναι: $n = 2$, $n_1 = 2$ και $n_2 = 2$. Συνδέοντας τα δεδομένων, που δίνονται στο Σχ.4.3.2 (συμπεριλαμβανομένων των μέγιστων ρυθμών παραγωγής της κανονικής περιόδου και των υπερωριών) σε αυτό το γενικό μοντέλο παίρνουμε το συγκεκριμένο μοντέλο, για αυτήν την εφαρμογή. Ειδικότερα, το νέο πρόβλημα γραμμικού προγραμματισμού είναι ο καθορισμός των τιμών των x_{1R}, x_{1O}, x_{2R} και x_{2O}, έτσι ώστε:

Μεγιστοποίησε: $Z = 3x_{1R} + 2x_{1O} + 5x_{2R} + x_{2O}$,

υπό τους περιορισμούς:

$$x_{1R} + x_{1O} \leq 4$$

$$2(x_{2R} + x_{2O}) \leq 12$$

$$3(x_{1R} + x_{1O}) + 2(x_{2R} + x_{2O}) \leq 18$$

$$x_{1R} \leq 3, x_{1O} \leq 1, x_{2R} \leq 3, x_{2O} \leq 3$$

και

$$x_{1R} \geq 0, x_{1O} \geq 0, x_{2R} \geq 0, x_{2O} \geq 0.$$

(Σημειώνουμε ότι οι περιορισμοί άνω φράγματος που είναι μετά την τελευταία σειρά του μοντέλου καθιστούν τους δύο πρώτους συναρτησιακούς περιορισμούς *περιττούς*, οπότε μπορούμε να διαγράψουμε αυτούς τους δύο περιορισμούς).

Ωστόσο, υπάρχει ένας σημαντικός παράγοντας που δεν λαμβάνεται υπόψη με σαφήνεια σε αυτήν τη διατύπωση (φορμαλισμό). Συγκεκριμένα, δεν υπάρχει τίποτα στο μοντέλο που να απαιτεί όλον το διαθέσιμο κανονικό χρόνο, για να χρησιμοποιηθεί πλήρως ένα προϊόν, πριν χρησιμοποιηθεί κάποια υπερωρία, για το προϊόν. Με άλλα λόγια, μπορεί να είναι εφικτό να έχουμε $x_{1O} > 0$ ακόμη και όταν $x_{1R} < 3$ και να έχουμε $x_{2O} > 0$ ακόμη και όταν $x_{2R} < 3$. Οι λύσεις αυτές δεν θα ήταν, ωστόσο, αποδεκτές από τη διαχείριση. (Η απαγόρευση τέτοιων λύσεων είναι ο ειδικός περιορισμός, που συζητήθηκε νωρίτερα σε αυτήν την ενότητα.)

Τώρα, ερχόμαστε στη *βασική ιδιότητα του διαχωρίσιμου προγραμματισμού*. Μπορεί το μοντέλο να μη λαμβάνει υπόψη τον παράγοντα αυτό άμεσα, αλλά τον έχει λάβει έμμεσα υπόψη! Παρά το γεγονός ότι η περίσσεια "εφικτών" λύσεων του μοντέλου, που στην πραγματικότητα είναι μη αποδεκτές, κάθε *βέλτιστη* λύση για το μοντέλο είναι *εγγυημένα* νόμιμη, η οποία δεν αντικαθιστά διαθέσιμη εργασία κανονικής απασχόλησης με υπερωριακή εργασία. Ως εκ τούτου, η μέθοδος Simplex μπορεί να εφαρμοστεί με ασφάλεια σε αυτό το μοντέλο, για να βρούμε την πιο κερδοφόρο αποδεκτή σύνθεση προϊόντων. Οι λόγοι είναι δύο. Πρώτον, οι δύο μεταβλητές απόφασης για κάθε προϊόν εμφανίζονται *πάντα* μαζί ως *άθροισμα*, $x_{1R} + x_{1O}$ ή $x_{2R} + x_{2O}$, σε *κάθε* συναρτησιακό περιορισμό, εκτός από το άνω φράγμα των περιορισμών στις επιμέρους μεταβλητές. Ως εκ τούτου, είναι *πάντα* δυνατό να μετατρέψουμε μία μη αποδεκτή εφικτή λύση σε μία αποδεκτή, η οποία έχει τον ίδιο συνολικό ρυθμό παραγωγής, $x_1 = x_{1R} + x_{1O}$ και $x_2 = x_{2R} + x_{2O}$, απλά αντικαθιστώντας την παραγωγή των υπερωριών με παραγωγή κανονικού χρόνου, όσο δυνατόν περισσότερο. Δεύτερον, η παραγωγή των υπερωριών είναι λιγότερο επικερδής από την παραγωγή κανονικού χρόνου (για παράδειγμα, η κλίση κάθε καμπύλης κέρδος στο Σχ.4.3.2 είναι μια γνησίως *φθίνουσα* συνάρτηση του ρυθμού παραγωγής), έτσι μετατρέποντας μία μη αποδεκτή εφικτή λύση σε μία αποδεκτή, με αυτόν τον τρόπο, πρέπει να αυξήσουμε το συνολικό ποσοστό κέρδους Z Κατά συνέπεια, κάθε εφικτή λύση που χρησιμοποιεί στην παραγωγή υπερωρίες, για ένα προϊόν που η παραγωγή κανονικού χρόνου είναι ακόμα διαθέσιμη, *δεν* μπορεί να είναι βέλτιστη σύμφωνα με το μοντέλο.

Για παράδειγμα, θεωρούμε την μη αποδεκτή εφικτή λύση $x_{1R} = 1$, $x_{1O} = 1$, $x_{2R} = 1$, $x_{2O} = 3$, η οποία δίνει συνολική τιμή κέρδους $Z = 13$. Ο μη αποδεκτός τρόπος για την επίτευξη του ίδιου συνόλου συντελεστών παραγωγής $x_1 = 2$ και $x_2 = 4$ είναι $x_{1R} = 2$, $x_{1O} = 0$, $x_{2R} = 3$, $x_{2O} = 1$. Αυτή η τελευταία λύση είναι εφικτή, αλλά επίσης αυξάνει το Z κατά $(3-2)(1) + (5-1)2 = 9$ σε συνολική τιμή κέρδους $Z = 22$.

Ομοίως, η βέλτιστη λύση για το μοντέλο αυτό προκύπτει ότι είναι $x_{1R} = 3$, $x_{1O} = 1$, $x_{2R} = 3$, $x_{2O} = 0$, η οποία είναι μια αποδεκτή εφικτή λύση.

Επεκτάσεις

Μέχρι στιγμής, έχουμε επικεντρωθεί στην ειδική περίπτωση διαχωρίσιμου προγραμματισμού, όπου η μόνη μη γραμμική συνάρτηση είναι η αντικειμενική συνάρτηση $f(x)$. Τώρα, εξετάζουμε τη γενική περίπτωση, όπου οι συναρτήσεις περιορισμών $g_i(x)$ δεν χρειάζεται να είναι γραμμικές, αλλά είναι κυρτές και διαχωρίσιμες, έτσι ώστε κάθε $g_i(x)$ να μπορεί να εκφραστεί ως άθροισμα των συναρτήσεων των επιμέρους μεταβλητών

$$g_i(x) = \sum_{j=1}^{n} g_{ij}(x_j),$$

όπου κάθε $g_{ij}(x_j)$ είνα κυρτή συνάρτηση. Για άλλη μια φορά, κάθε μία από αυτές τις νέες συναρτήσεις μπορεί να προσεγγίζεται, όσο είναι επιθυμητό, από μία τμηματικά γραμμική συνάρτηση (αν δεν είναι ήδη σε αυτή τη μορφή). Ένας νέος περιορισμός είναι ότι για κάθε μεταβλητή x_j $(j = 1, 2, \ldots, n)$, όλες οι τμηματικά γραμμικές προσεγγίσεις των συναρτήσεων αυτής της μεταβλητής, $[f_j(x_j), g_{1j}(x_j), \ldots, g_{mj}(x_j)]$, πρέπει να έχουν τα *ίδια* σημεία διακοπής, ώστε οι ίδιες νέες μεταβλητές $(x_{j1}, x_{j2}, \ldots, x_{jn_j})$ να μπορούν να χρησιμοποιηθούν για όλες τις τμηματικά γραμμικές συναρτήσεις. Αυτός ο φορμαλισμός οδηγεί σε ένα μοντέλο γραμμικού προγραμματισμού, ακριβώς όπως αυτό που δόθηκε για την ειδική περίπτωση, εκτός από το ότι για κάθε i και j, οι μεταβλητές x_{jk} έχουν διαφορετικούς συντελεστές στον περιορισμό i [όπου οι συντελεστές αυτοί είναι οι αντίστοιχες κλίσεις της τμηματικά γραμμικής συνάρτησης, που προσεγγίζει την $g_{ij}(x_j)$]. Επειδή, η $g_{ij}(x_j)$ απαιτείται να είναι κυρτή, η ίδια λογική όπως και πριν συνεπάγεται ότι εξακολουθεί να ισχύει η βασική ιδιότητα του διαχωρίσιμου προγραμματισμού.

Ένα μειονέκτημα της προσέγγισης των συναρτήσεων από τμηματικά γραμμικές συναρτήσεις, όπως περιγράφεται σε αυτή την ενότητα, είναι ότι η επίτευξη μιας καλής προσέγγισης απαιτεί ένα μεγάλο αριθμό ευθύγραμμων τμημάτων (μεταβλητών), ενώ ένα τέτοιο πλέγμα σφαλμάτων για τα σημεία διακοπής είναι αναγκαίο, μόνο στη γειτονιά της βέλτιστης λύσης. Ως εκ τούτου, έχουν αναπτυχθεί πιο εξελιγμένες προσεγγίσεις, που χρησιμοποιούν μία ακολουθία τμηματικά γραμμικών συναρτήσεων *δύο τμημάτων*, για να ληφθούν *προοδευτικά στενότερες προσεγγίσεις*, με άμεση γειτνίαση. Αυτού του είδους η προσέγγιση τείνει να είναι ταχύτερη και ακριβέστερη, στην προσέγγιση της βέλτιστης λύσης.

4.4 Κυρτός Προγραμματισμός

Έχουμε ήδη συζητήσει κάποιες ειδικές περιπτώσεις κυρτού προγραμματισμού, στις Εν.3.1-3.2 (Κεφάλαιο 3) (προβλήματα χωρίς περιορισμούς), στην Εν.4.2 (τετραγωνική αντικειμενική συνάρτηση, με γραμμικούς περιορισμούς) και στην Εν.4.3 (διαχωρίσιμες συναρτήσεις). Επίσης, έχουμε δει τη θεωρία για τη γενική περίπτωση (αναγκαίες και ικανές συνθήκες για βελτιστότητα), στην Εν.4.1. Σε αυτή την ενότητα, θα συζητήσουμε, εν συντομία, ορισμένα είδη των προσεγγίσεων που χρησιμοποιούνται, για την επίλυση του γενικότερου προβλήματος κυρτού προγραμματισμού [όπου η αντικειμενική συνάρτηση προς μεγιστοποίηση, $f(x)$, είναι κοίλη και οι συναρτήσεις περιορισμών, $g_i(x)$, είναι κυρτές], και στη συνέχεια, παρουσιάζουμε ένα παράδειγμα αλγόριθμου για κυρτό προγραμματισμό.

Δεν υπάρχει ενιαίος αλγόριθμος, που να χρησιμοποιείται πάντα, για την επίλυση προβλημάτων κυρτού προγραμματισμού. Έχουν αναπτυχθεί πολλοί διαφορετικοί αλγόριθμοι, ο καθένας με τα δικά του πλεονεκτήματα και μειονεκτήματα και εξακολουθεί να διενεργείται έρευνα σε αυτόν τον τομέα. Σε γενικές γραμμές, οι περισσότεροι από αυτούς τους αλγορίθμους εμπίπτουν σε μία από τις ακόλουθες τρεις κατηγορίες.

Η πρώτη κατηγορία είναι οι **αλγόριθμοι κλίσης**, όπου η διαδικασία αναζήτησης μέσω κλίσης της Εν.3.2 τροποποιείται κατά κάποιο τρόπο, για να διαφυλάξει τη διαδικασία αναζήτησης από τη διείσδυση κάθε περιοριστικού φράγματος. Για παράδειγμα, μία δημοφιλής μέθοδος κλίσης είναι η μέθοδος *γενικευμένα μειωμένης κλίσης* (ΓΜΒ).

Η δεύτερη κατηγορία -**ακολουθιακοί αλγόριθμοι, χωρίς περιορισμούς**- περιλαμβάνει *συνάρτηση ποινής* και μεθόδους *συνάρτησης φράγματος*. Αυτοί οι αλγόριθμοι μετατρέπουν το αρχικό πρόβλημα βελτιστοποίησης, με περιορισμούς, σε μια ακολουθία *προβλημάτων βελτιστοποίησης, χωρίς περιορισμούς*, των οποίων οι βέλτιστες λύσεις συγκλίνουν στη βέλτιστη λύση του αρχικού προβλήματος. Κάθε ένα από αυτά τα προβλήματα βελτιστοποίησης, χωρίς περιορισμούς, μπορούν να επιλυθούν με τη διαδικασία αναζήτησης μέσω κλίσης, της Εν.3.2. Η μετατροπή αυτή επιτυγχάνεται με την ενσωμάτωση των περιορισμών σε μία συνάρτηση ποινής (ή συνάρτηση φράγματος), που αφαιρείται από την αντικειμενική συνάρτηση, προκειμένου να επιβάλλει μεγάλες ποινές για την παραβίαση περιορισμών (ή ακόμη και όταν είναι κοντά στα όρια περιορισμού). Θα δούμε ένα παράδειγμα αυτής της κατηγορίας αλγορίθμων, στην επόμενη ενότητα.

Η τρίτη κατηγορία -**αλγόριθμοι ακολουθιακής προσέγγισης**- περιλαμβάνει μεθόδους *γραμμικής* και *τετραγωνικής προσέγγισης*. Αυτοί οι αλγόριθμοι αντικαθιστούν τη μη γραμμική αντικειμενική συνάρτηση με μια ακολουθία γραμμικών ή τετραγωνικών προσεγγίσεων. Για προβλήματα βελτιστοποίησης, με γραμμικούς περιορισμούς, οι προσεγγίσεις αυτές επιτρέπουν την επανειλημμένη εφαρμογή αλγορίθμων γραμμικού ή τετραγωνικού προγραμματισμού. Η εργασία αυτή συνοδεύεται από άλλες αναλύσεις, που παράγουν μια ακολουθία λύσεων, που συγκλίνουν σε μια βέλτιστη λύση, για το αρχικό πρόβλημα. Παρά το γεγονός, ότι αυτοί οι αλγόριθμοι είναι ιδιαίτερα κατάλληλοι για προβλήματα βελτιστοποίησης, με γραμμικούς περιορισμούς, μερικοί μπορούν επίσης να επεκταθούν σε προβλήματα με μη γραμμικές συναρτήσεις περιορισμών, με τη χρήση των κατάλληλων γραμμικών προσεγγίσεων.

Ως ένα παράδειγμα ενός αλγορίθμου *ακολουθιακής προσέγγισης*, παρουσιάζουμε τον **αλγόριθμο Frank-Wolfe**, για την περίπτωση *γραμμικά περιορισμένου* κυρτού προγραμματισμού (οπότε, οι περιορισμοί είναι $\boldsymbol{Ax} \leq \boldsymbol{b}$ και $\boldsymbol{x} \geq \boldsymbol{0}$, σε μητρωική μορφή). Αυτή η διαδικασία είναι ιδιαίτερα απλή, αφού συνδυάζει *γραμμικές* προσεγγίσεις της αντικειμενικής συνάρτησης (δίνοντάς μας τη δυνατότητα να χρησιμοποιήσουμε τη μέθοδο Simplex), με τη μονοδιάστατη διαδικασία αναζήτησης της Εν.3.1.

Ένας ακολουθιακός αλγόριθμος γραμμικής προσέγγισης

(Frank-Wolfe)

Δεδομένης μιας ενδιάμεσης εφικτής λύσης $\boldsymbol{x}'$, η γραμμική προσέγγιση, που χρησιμοποιείται για την αντικειμενική συνάρτηση $f(x)$

είναι η πρώτης τάξης σειρά Taylor, επέκταση της $f(x)$, γύρω από το $\boldsymbol{x} = \boldsymbol{x}'$, δηλαδή:

$$f(\boldsymbol{x}') \approx f(\boldsymbol{x}') + \sum_{j=1}^{n} \frac{\partial f(x')}{\partial x_j}\left(x_j - x_j'\right) = f(\boldsymbol{x}') + \nabla f(\boldsymbol{x}')(\boldsymbol{x} - \boldsymbol{x}'),$$

όπου οι μερικές παράγωγοι υπολογίζονται στο $\boldsymbol{x} = \boldsymbol{x}'$. Επειδή, οι $f(\boldsymbol{x}')$ και $\nabla f(\boldsymbol{x}')\boldsymbol{x}'$ έχουν σταθεροποιημένες τιμές, μπορούν να απλοποιηθούν, για να πάρουμε μια ισοδύναμη γραμμική αντικειμενική συνάρτηση:

$$g(\boldsymbol{x}) = \nabla f(\boldsymbol{x}')\boldsymbol{x} = \sum_{j=1}^{n} c_j x_j, \text{ όπου } c_j = \frac{\partial f(x)}{\partial x_j} \text{ στο } \boldsymbol{x} = \boldsymbol{x}'.$$

Η μέθοδος Simplex (ή η γραφική διαδικασία, για $n = 2$), τότε εφαρμόζεται στο προκύπτον πρόβλημα γραμμικού προγραμματισμού [μεγιστοποίηση της $g(\boldsymbol{x})$ υποκείμενη στους αρχικούς περιορισμούς, $\boldsymbol{Ax} \leq \boldsymbol{b}$ και $\boldsymbol{x} \geq \boldsymbol{0}$] για να βρούμε τη βέλτιστη λύση του, $\boldsymbol{x_{LP}}$. Σημειώνουμε ότι η γραμμική αντικειμενική συνάρτηση αναγκαστικά αυξάνει σταθερά καθώς κινούμαστε κατά μήκος του ευθύγραμμου τμήματος, από το $\boldsymbol{x}'$ ως το $\boldsymbol{x_{LP}}$ (το οποίο είναι στο σύνορο της εφικτής περιοχής). Ωστόσο, η γραμμική προσέγγιση μπορεί να μην είναι ιδιαίτερα κοντά, για $\boldsymbol{x}$ μακριά από το $\boldsymbol{x}'$, ώστε η *μη γραμμική* αντικειμενική συνάρτηση να μην μπορεί να συνεχίσει να αυξάνει, σε όλη τη διαδρομή από το $\boldsymbol{x}'$ ως το $\boldsymbol{x_{LP}}$. Η εκ τούτου, παρά να αποδεχτούμε την $\boldsymbol{x_{LP}}$ ως την επόμενη ενδιάμεση λύση, επιλέγουμε το σημείο που μεγιστοποιεί τη μη γραμμική αντικειμενική συνάρτηση, κατά μήκος αυτού ευθύγραμμου τμήματος. Αυτό το στοιχείο μπορεί να βρεθεί μέσω της *μονοδιάστατης διαδικασίας αναζήτησης*, της Εν.3.1, όπου η μία μεταβλητή, για τους σκοπούς αυτής της αναζήτησης, είναι το κλάσμα t της συνολικής απόστασης από το $\boldsymbol{x}'$ ως το $\boldsymbol{x_{LP}}$. Το στοιχείο αυτό γίνεται τότε η νέα ενδιάμεση λύση, για την έναρξη της επόμενης επανάληψης του αλγόριθμου, όπως περιγράψαμε. Η ακολουθία των ενδιάμεσων λύσεων, που παράγονται από τις επαναλαμβανόμενες επαναλήψεις συγκλίνει στη βέλτιστη λύση του αρχικού προβλήματος, οπότε ο αλγόριθμος σταματά μόλις οι επόμενες ενδιάμεσες λύσεις είναι αρκετά κοντά μεταξύ τους, για να έχουν φτάσει στη βέλτιστη λύση.

Περίληψη του Αλγόριθμου Frank-Wolfe.

Αρχικοποίηση: Βρίσκουμε μια εφικτή αρχική ενδιάμεση λύση $\boldsymbol{x}^{(0)}$, για παράδειγμα, με την εφαρμογή των διαδικασιών γραμμικού προγραμματισμού για να βρούμε μια αρχική βασική εφικτή λύση. Έστω, $k = 1$.

Επαναλήψεις:

1. Για $j = 1, 2, \ldots, n$ υπολογίζουμε την ποσότητα:

$$\frac{\partial f(\boldsymbol{x})}{\partial x_j}, \text{ στο } \boldsymbol{x} = \boldsymbol{x}^{(\boldsymbol{k-1})}$$

και θέτουμε c_j ίσο με αυτήν την ποσότητα.

2. Βρίσκουμε μία βέλτιστη λύση $\boldsymbol{x}_{\boldsymbol{LP}}^{(\boldsymbol{k})}$, για το ακόλουθο πρόβλημα γραμμικού προγραμματισμού.

Μεγιστοποίησε: $g(x) = \sum_{j=1}^{n} c_j x_j$,

υπό τους περιορισμούς:

$\boldsymbol{Ax} \leq \boldsymbol{b}$ και $\boldsymbol{x} \geq \boldsymbol{0}$.

3. Για τη μεταβλητή t $(0 \leq t \leq 1)$, θέτουμε:

$$h(t) = f(\boldsymbol{x}), \text{ για } \boldsymbol{x} = \boldsymbol{x}^{(\boldsymbol{k-1})} + t(\boldsymbol{x}_{\boldsymbol{LP}}^{(\boldsymbol{k})} - \boldsymbol{x}^{(\boldsymbol{k-1})}),$$

έτσι ώστε $h(t)$ να δίνει την τιμή της $f(\boldsymbol{x})$ στο ευθύγραμμο τμήμα μεταξύ $\boldsymbol{x}^{(\boldsymbol{k-1})}$ (όπου το $t = 0$) και $\boldsymbol{x}_{\boldsymbol{LP}}^{(\boldsymbol{k})}$ (όπου το $t = 1$). Χρησιμοποιούμε κάποια διαδικασία, όπως η μονοδιάστατη διαδικασία αναζήτησης (βλέπε Εν.3.1) για να μεγιστοποιήσουμε την $h(t)$ στο διάστημα $0 \leq t \leq 1$ και θέτουμε $\boldsymbol{x}^{(\boldsymbol{k})}$ ίσο με το αντίστοιχο $\boldsymbol{x}$. Πηγαίνουμε στο κριτήριο διακοπής.

Κριτήριο διακοπής: Αν $\boldsymbol{x}^{(\boldsymbol{k-1})}$ και $\boldsymbol{x}^{(\boldsymbol{k})}$ είναι αρκετά κοντά, τότε σταματάμε και χρησιμοποιούμε το $\boldsymbol{x}^{(\boldsymbol{k})}$ (ή κάποια προέκτασή του, $\boldsymbol{x}^{(\boldsymbol{0})}, \boldsymbol{x}^{(\boldsymbol{1})}, \ldots, \boldsymbol{x}^{(\boldsymbol{k-1})}, \boldsymbol{x}^{(\boldsymbol{k})}$), όπως εκτίμηση της βέλτιστης λύσης. Σε αντίθετη περίπτωση, επαναφέρουμε $k = k + 1$ και εκτελούμε επόμενη επανάληψη.

Τώρα ας παρουσιάσουμε αυτήν τη διαδικασία.

Παράδειγμα. Θεωρούμε το ακόλουθο γραμμικά περιορισμένο πρόβλημα κυρτού προγραμματισμού:

Μεγιστοποίησε: $f(\boldsymbol{x}) = 5x_1 - x_1^2 + 8x_2 - 2x_2^2$,

υπό τους περιορισμούς:

$$3x_1 + 2x_2 \leq 6$$

και

$$x_1 \geq 0, x_2 \geq 0.$$

Σημειώνουμε ότι:

$$\frac{\partial f}{\partial x_1} = 5 - 2x_1, \frac{\partial f}{\partial x_2} = 8 - 4x_2,$$

έτσι ώστε το *μη περιορισμένο* μέγιστο $\boldsymbol{x} = (\frac{5}{2}, 2)$ να παραβιάζει το συναρτησιακό περιορισμό. Έτσι, χρειάζεται περισσότερη δουλειά για να βρούμε το *περιορισμένο* μέγιστο.

Επειδή, η $\boldsymbol{x} = (0{,}0)$ είναι σαφώς εφικτή (και αντιστοιχεί στην αρχική βασική εφικτή λύση, για τους περιορισμούς γραμμικού προγραμματισμού), ας την επιλέξουμε ως την αρχική ενδιάμεση λύση $\boldsymbol{x}^{(\mathbf{0})}$, για τον αλγόριθμο Frank-Wolfe. Αντικαθιστώντας τις $x_1 = 0$ και $x_2 = 0$ στις, εκφράσεις των μερικών παραγώγων, παίρνουμε $c_1 = 5$ και $c_2 = 8$, έτσι ώστε $g(\boldsymbol{x}) = 5x_1 + 8x_2$ είναι η αρχική γραμμική προσέγγιση της αντικειμενικής συνάρτησης. Η γραφική επίλυση αυτού του προβλήματος γραμμικού προγραμματισμού (βλέπε Σχ.4.4.1(α)) δίνει $\boldsymbol{x}_{\boldsymbol{LP}}^{(\mathbf{1})} = (0{,}3)$. Για το βήμα 3 της πρώτης επανάληψης, τα σημεία στο ευθύγραμμο τμήμα, μεταξύ (0,0) και (0,3), που φαίνονται στο Σχ.4.4.1(α) εκφράζονται ως εξής:

$$(x_1, x_2) = (0{,}0) + t[(0{,}3) - (0{,}0)], \text{ για } 0 \leq t \leq 1$$

$$= (0{,}3t)$$

όπως φαίνεται στην έκτη στήλη του Πίνακα 4.4.Ι. Αυτή η έκφραση δίνει:

$$h(t) = f(0{,}3t) = 8(3t) - 2(3t)^2$$

$$= 24t - 18t^2,$$

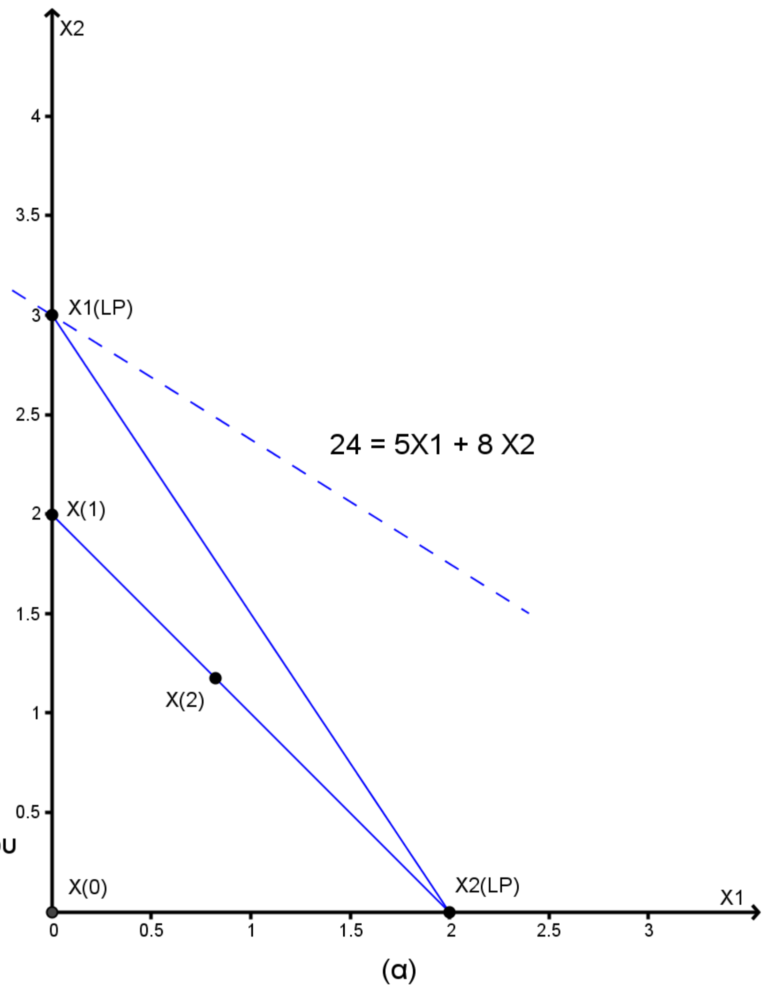
έτσι ώστε η τιμή $t = t^*$, που μεγιστοποιεί την $h(t)$, για $0 \leq t \leq 1$, μπορεί να ληφθεί σε αυτήν την περίπτωση, θέτοντας:

$$\frac{dh(t)}{dt} = 24 - 36t = 0,$$

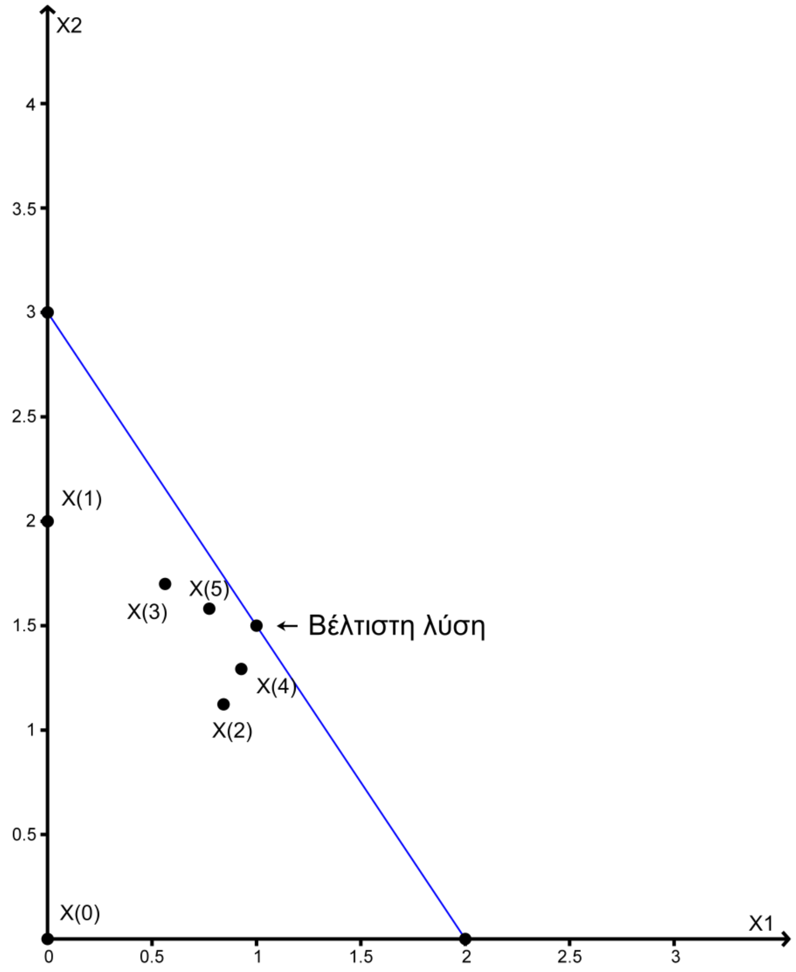
έτσι, ώστε $t^* = \frac{2}{3}$. Αυτό το αποτέλεσμα οδηγεί στην παρακάτω ενδιάμεση λύση:

$$\begin{aligned} \boldsymbol{x}^{(\mathbf{1})} &= (0{,}0) + \tfrac{2}{3}[(0{,}3) - (0{,}0)] \\ &= (0{,}2), \end{aligned}$$

η οποία ολοκληρώνει την πρώτη επανάληψη.



Σχήμα 4.4.1
Παρουσίαση του αλγόριθμου Frank-Wolfe



(β)

K	$\boldsymbol{x^{(k-1)}}$	$\boldsymbol{c_1}$	$\boldsymbol{c_2}$	$\boldsymbol{x_{LP}^{(k)}}$	$\boldsymbol{x}$ **στην** $\boldsymbol{h(t)}$	$\boldsymbol{h(t)}$	$\boldsymbol{t^*}$	$\boldsymbol{x^{(k)}}$
1	(0,0)	5	8	(0,3)	$(0,3t)$	$24t - 18t^2$	$\frac{2}{3}$	(0,2)
2	(0,2)	5	0	(2,0)	$(2t, 2-2t)$	$8 + 10 - 12t^2$	$\frac{5}{12}$	$(\frac{5}{6}, \frac{7}{6})$

Πίνακας 4.4.I Εφαρμογή του αλγόριθμου Frank-Wolfe στο παράδειγμα.

Για να σχεδιάσουμε τους υπολογισμούς που οδηγούν στα αποτελέσματα της δεύτερης σειράς του Πίνακα 4.4.I, η σχέση $\boldsymbol{x}^{(\mathbf{1})} = (0,2)$ δίνει:

$$c_1 = 5 - 2(0) = 5,$$

$$c_2 = 8 - 4(2) = 0.$$

Για την αντικειμενική συνάρτηση $g(\boldsymbol{x}) = 5x_1$, η γραφική επίλυση του προβλήματος στην εφικτή περιοχή του Σχ.4.4.1(α) δίνει $\boldsymbol{x}_{\boldsymbol{LP}}^{(\mathbf{2})} = (2,0)$. Ως εκ τούτου, η έκφραση για το ευθύγραμμο τμήμα, μεταξύ των $\boldsymbol{x}^{(\mathbf{1})}$ και $\boldsymbol{x}_{\boldsymbol{LP}}^{(\mathbf{2})}$ (βλέπε Σχ.4.4.1(α)) είναι:

$$\begin{aligned} \boldsymbol{x} &= (0,2) + t[(2,0) - (0,2)] \\ &= (2t, 2 - 2t), \end{aligned}$$

έτσι, ώστε:

$$\begin{aligned} h(t) &= f(2t, 2 - 2t) \\ &= 5(2t) - (2t)^2 + 8(2 - 2t) - 2(2 - 2t)^2 \\ &= 8 + 10t - 12t^2. \end{aligned}$$

Θέτοντας:

$$\frac{dh(t)}{dt} = 10 - 24t = 0$$

παίρνουμε $t^* = \frac{5}{12}$. Ως εκ τούτου:

$$\begin{aligned} \boldsymbol{x}^{(\mathbf{2})} &= (0,2) + \frac{5}{12}[(0,2) - (0,2)] \\ &= (\frac{5}{6}, \frac{7}{6}). \end{aligned}$$

Μπορούμε να δούμε στο Σχ.4.4.1(β) πώς οι ενδιάμεσες λύσεις εναλλάσσονται μεταξύ δύο τροχιών, που φαίνεται να διασταυρώνονται στο σημείο $\boldsymbol{x} = (1, \frac{3}{2})$. Αυτό το σημείο είναι, πράγματι, η βέλτιστη λύση, όπως μπορεί να επαληθευτεί, με την εφαρμογή των συνθηκών ΚΚΤ της Εν.4.1.

Αυτό το παράδειγμα απεικονίζει ένα κοινό χαρακτηριστικό των αλγορίθμων Frank-Wolfe, δηλαδή, ότι οι ενδιάμεσες λύσεις εναλλάσσονται μεταξύ δύο (ή περισσοτέρων) τροχιών. Όταν εναλλάσσονται με αυτόν τον τρόπο, μπορούμε να προεκτείνουμε τις τροχιές στο κατά προσέγγιση σημείο τομής τους, ώστε να εκτιμήσουμε μια βέλτιστη λύση. Η εκτίμηση αυτή τείνει να είναι καλύτερη από τη χρήση της τελευταίας ενδιάμεσης λύσης που παράγουμε. Ο λόγος είναι ότι οι ενδιάμεσες λύσεις τείνουν να συγκλίνουν μάλλον αργά προς μία βέλτιστη λύση, ώστε η τελευταία ενδιάμεση λύση μπορεί να εξακολουθεί να είναι αρκετά μακριά από τη βέλτιστη.

Εν κατακλείδι, τονίζουμε ότι ο αλγόριθμος Frank-Wolfe είναι μόνο ένα παράδειγμα των αλγορίθμων ακολουθιακής προσέγγισης. Πολλοί από αυτούς τους αλγορίθμους χρησιμοποιούν *τετραγωνική*, αντί της *γραμμικής* προσέγγισης, σε κάθε επανάληψη, επειδή η τετραγωνική προσέγγιση παρέχει μια σημαντικά καλύτερη προσαρμογή στο αρχικό πρόβλημα και έτσι επιτρέπουν την ακολουθία των λύσεων να συγκλίνει σημαντικά ταχύτερα προς μια βέλτιστη λύση από ό,τι στην περίπτωση του Σχ.4.4.1(β). Για το λόγο αυτό, παρόλο που οι μέθοδοι ακολουθιακής γραμμικής προσέγγισης, όπως ο αλγόριθμος Frank-Wolfe, είναι σχετικά απλές στη χρήση, προτιμώνται οι *μέθοδοι ακολουθιακής τετραγωνικής προσέγγισης*, γενικά, σε πραγματικές εφαρμογές. Δημοφιλείς μεταξύ αυτών είναι οι *quasi-Newton* (ή *παραμετρικές*) μέθοδοι, οι οποίες υπολογίζουν μια τετραγωνική προσέγγιση της καμπυλότητας μιας μη γραμμικής συνάρτησης, χωρίς τον ακριβή υπολογισμό δεύτερων (μερικών) παραγώγων. (Για γραμμικά περιορισμένα προβλήματα βελτιστοποίησης, αυτή η μη γραμμική συνάρτηση είναι η αντικειμενική συνάρτηση, ενώ για προβλήματα με μη γραμμικούς περιορισμούς, είναι μία Lagrangian συνάρτηση.) Μερικοί quasi-Newton αλγόριθμοι δεν σχηματίζουν και επιλύουν ένα προσεγγιστικό τετραγωνικό πρόβλημα προγραμματισμού σε κάθε επανάληψη, αλλά, αντίθετα, ενσωματώνουν ορισμένα από τα βασικά συστατικά των *αλγόριθμων κλίσης*.

Ορισμένες επιλογές Λογισμικού

Όπως υποδεικνύεται στο τέλος της Εν.4.2, τόσο το Excel, όσο και το LINGO μπορούν να λύσουν προβλήματα κυρτού προγραμματισμού, αλλά τα LINDO και CPLEX δεν μπορούν, εκτός από την ειδική περίπτωση του τετραγωνικού προγραμματισμού (η οποία περιλαμβάνει το παράδειγμα αυτής της ενότητας). Η επαγγελματική έκδοση του MPL υποστηρίζει ένα μεγάλο αριθμό *solver*, συμπεριλαμβανομένων και ορισμένων που μπορούν να χειριστούν κυρτό προγραμματισμό. Ένας από αυτούς, που ονομάζεται CONOPT, περιλαμβάνεται στην έκδοση για φοιτητές του MPL, που βρίσκεται σε CD-ROM. Τα παραδείγματα κυρτού προγραμματισμού, που διατυπώνονται στο αρχείο MPL αυτού του κεφαλαίου, έχουν λυθεί με αυτόν τον solver, επιλέγοντας *Μη γραμμικά μοντέλα*, για την εισαγωγή *Προεπιλεγμένου Τύπου Μοντέλου*, στο παράθυρο διαλόγου επιλογής MPL Γλώσσας.

4.5 Μη κυρτός Προγραμματισμός

Οι παραδοχές του κυρτού προγραμματισμού είναι πολύ βολικές, επειδή εξασφαλίζουν ότι κάθε *τοπικό μέγιστο* είναι και *ολικό*. Δυστυχώς, τα μη γραμμικά προβλήματα προγραμματισμού, που ανακύπτουν στην πράξη, συχνά έρχονται μόνο αρκετά κοντά στο να ικανοποιούν αυτές τις υποθέσεις, αλλά έχουν κάποιες σχετικά μικρές διαφορές. Τι είδους προσέγγιση μπορεί να χρησιμοποιηθεί, όταν ασχολούμαστε τέτοια προβλήματα *μη κυρτού προγραμματισμού*;

Μια κοινή προσέγγιση είναι η εφαρμογή μιας αλγοριθμικής *διαδικασίας αναζήτησης*, που θα σταματάει όταν βρίσκει ένα *τοπικό μέγιστο* και στη συνέχεια θα κάνει επανεκκίνηση αρκετές φορές από διάφορες αρχικές ενδιάμεσες λύσεις, για να βρει όσα διαφορετικά τοπικά μέγιστα είναι δυνατό. Το καλύτερο από αυτά τα τοπικά μέγιστα επιλέγεται, στη συνέχεια, για την εκτέλεση. Κανονικά, η διαδικασία αναζήτησης είναι αυτή που έχει σχεδιαστεί για να καταλήξουμε σε ολικό μέγιστο, όταν ικανοποιούνται όλες οι υποθέσεις του κυρτού προγραμματισμού, αλλά επίσης μπορεί να λειτουργήσει για να βρούμε ένα τοπικό μέγιστο, όταν δεν ικανοποιούνται.

Μια τέτοια διαδικασία αναζήτησης, που έχει χρησιμοποιηθεί ευρέως από την ανάπτυξή της στη δεκαετία του 1960, είναι η *τεχνική ακολουθιακής ελαχιστοποίησης, χωρίς περιορισμούς* (*sequential unconstrained minimization technique* ή SUMT για συντομία). Υπάρχουν, στην πραγματικότητα δύο κύριες εκδοχές της SUMT, μία εκ των οποίων είναι ένας αλγόριθμος *εξωτερικού σημείου*, που ασχολείται με μη εφικτές λύσεις, χρησιμοποιώντας μια *συνάρτηση ποινής*, για να αναγκάσει τη σύγκλιση στην εφικτή περιοχή. Θα περιγράψουμε την άλλη εκδοχή, η οποία είναι ένας αλγόριθμος *εσωτερικού σημείου*, που ασχολείται κατ’ ευθείαν με εφικτές λύσεις, χρησιμοποιώντας μια *συνάρτηση φραγμού*, για να αναγκάσει την παραμονή τους στο εσωτερικό της εφικτής περιοχής. Παρόλο που η SUMT αρχικά παρουσιάστηκε ως μια τεχνική ελαχιστοποίησης, θα τη μετατρέψουμε σε μια τεχνική μεγιστοποίησης, ώστε να είναι συνεπής με το υπόλοιπο του κεφαλαίου. Ως εκ τούτου, θα συνεχίσουμε να υποθέτουμε ότι το πρόβλημα είναι στη μορφή, που δόθηκε στην αρχή του κεφαλαίου και ότι όλες οι συναρτήσεις είναι διαφορίσιμες.

Sequential Unconstrained Minimization Technique (SUMT)

(Τεχνική ακολουθιακής ελαχιστοποίησης, χωρίς περιορισμούς)

Όπως υποδηλώνει το όνομά της, η SUMT αντικαθιστά το αρχικό πρόβλημα με μια *ακολουθία μη περιορισμένων* προβλημάτων βελτιστοποίησης, των οποίων οι λύσεις *συγκλίνουν* σε μία λύση (τοπικό μέγιστο) του αρχικού προβλήματος. Η προσέγγιση αυτή είναι πολύ ελκυστική, γιατί τα προβλήματα βελτιστοποίησης χωρίς περιορισμούς είναι πολύ πιο εύκολο να λυθούν (βλέπε τη διαδικασία αναζήτησης μέσω κλίσης της Εν.3.2), σε σχέση με αυτά που έχουν περιορισμούς. Κάθε ένα από τα μη περιορισμένα προβλήματα σε αυτήν την ακολουθία περιλαμβάνει την επιλογή μιας (προοδευτικά μικρότερης) αυστηρά θετικής τιμής ενός βαθμωτού μεγέθους r και στη συνέχεια, την επίλυση ως προς $\boldsymbol{x}$, έτσι ώστε:

Μεγιστοποίησε: $P(\boldsymbol{x};r) = f(\boldsymbol{x}) - rB(\boldsymbol{x})$.

Εδώ η $B(\boldsymbol{x})$ είναι μια **συνάρτηση φραγμού**, που έχει τις ακόλουθες ιδιότητες (για εφικτές τιμές του $\boldsymbol{x}$**,** για το αρχικό πρόβλημα):

1. Η $B(\boldsymbol{x})$ είναι *μικρή*, όταν το $\boldsymbol{x}$ είναι *μακριά* από το σύνορο της εφικτής περιοχής.

2. Η $B(\boldsymbol{x})$ είναι *μεγάλη*, όταν το $\boldsymbol{x}$ είναι *κοντά* στο σύνορο της εφικτής περιοχής.

3. $B(\boldsymbol{x}) \to \infty$, καθώς η απόσταση από την (πλησιέστερο) σύνορο της εφικτής περιοχής $\to 0$.

Έτσι, ξεκινώντας τη διαδικασία αναζήτησης με μια *εφικτή* αρχική ενδιάμεση λύση και στη συνέχεια προσπαθώντας να αυξήσουμε την $P(\boldsymbol{x};r)$, η $B(\boldsymbol{x})$ παρέχει ένα *φράγμα*, που εμποδίζει την αναζήτηση από τη διέλευση (ή να φτάσει) το σύνορο της εφικτής περιοχής, για το αρχικό πρόβλημα.

Η πιο κοινή επιλογή της $B(\boldsymbol{x})$ είναι:

$$B(\boldsymbol{x}) = \sum_{i=1}^{m} \frac{1}{b_i - g_i(x)} + \sum_{j=1}^{n} \frac{1}{x_j}.$$

Για εφικτές τιμές του $\boldsymbol{x}$, σημειώνουμε ότι ο παρονομαστής κάθε όρου είναι ανάλογος της απόστασης του $\boldsymbol{x}$ από το όριο του περιορισμού, για τον αντίστοιχο συναρτησιακό περιορισμό ή περιορισμό μη αρνητικότητας. Κατά συνέπεια, *κάθε* όρος είναι ένας *όρος άπωσης συνόρου*, που έχει όλες τις προηγούμενες τρεις ιδιότητες, σε σχέση με το συγκεκριμένο όριο του περιορισμού. Ένα άλλο ελκυστικό χαρακτηριστικό της $B(\boldsymbol{x})$ είναι ότι όταν πληρούνται όλες οι προϋποθέσεις του *κυρτού προγραμματισμού*, η $P(\boldsymbol{x}; r)$ είναι *κοίλη* συνάρτηση.

Επειδή, η $B(\boldsymbol{x})$ διατηρεί την αναζήτηση μακριά από το σύνορο της εφικτής περιοχής, είναι πιθανό να δημιουργηθεί το πολύ εύλογο ερώτημα: Τι θα συμβεί αν η επιθυμητή λύση βρίσκεται εκεί; Αυτή η ανησυχία είναι ο λόγος που η SUMT περιλαμβάνει την επίλυση μιας σειράς τέτοιων προβλημάτων βελτιστοποίησης, χωρίς περιορισμούς, για διαδοχικά μικρότερες τιμές του r, που πλησιάζουν το μηδέν (όπου η τελική ενδιάμεση λύση από κάθε ένα γίνεται η αρχική ενδιάμεση λύση, για το επόμενο). Για παράδειγμα, κάθε νέο r θα μπορούσε να ληφθεί από το προηγούμενο με πολλαπλασιασμό με μια σταθερά θ $(0 < \theta < 1)$, όπου μια τυπική τιμή είναι $\theta = 0.01$. Καθώς το r πλησιάζει το 0, η $P(\boldsymbol{x}; r)$ προσεγγίζει την $f(x)$, οπότε το αντίστοιχο τοπικό μέγιστο της $P(\boldsymbol{x}; r)$ συγκλίνει σε ένα τοπικό μέγιστο του αρχικού προβλήματος. Ως εκ τούτου, είναι απαραίτητο να λυθούν αρκετά προβλήματα βελτιστοποίησης χωρίς περιορισμούς, ώστε να επιτραπεί η προέκταση των λύσεών τους σε αυτήν την περιοριστική λύση.

Πόσα είναι αρκετά, για να επιτρέψουν αυτήν την παρέκταση; Όταν το αρχικό πρόβλημα ικανοποιεί τις υποθέσεις του κυρτού προγραμματισμού, είναι διαθέσιμες κάποιες χρήσιμες πληροφορίες, για να μας καθοδηγήσουν σε αυτή την απόφαση. Συγκεκριμένα, εάν το $\overline{x}$ μεγιστοποιεί ολικά την $P(\boldsymbol{x}; r)$, τότε:

$$f(\overline{\boldsymbol{x}}) \leq f(\boldsymbol{x}^*) \leq f(\overline{\boldsymbol{x}}) + rB(\overline{\boldsymbol{x}}),$$

όπου το $\boldsymbol{x}^*$ είναι η (άγνωστη) *βέλτιστη* λύση, για το αρχικό πρόβλημα. Έτσι, το $rB(\overline{\boldsymbol{x}})$ είναι το *μέγιστο σφάλμα* (στην τιμή της αντικειμενικής συνάρτησης), που μπορεί να οδηγήσει, με τη χρήση του $\overline{\boldsymbol{x}}$, στην προσέγγιση του $\boldsymbol{x}^*$, και προεκτείνοντας πέρα από το $\overline{\boldsymbol{x}}$, για την αύξηση της $f(x)$ μειώνει περισσότερο το σφάλμα. Αν η *ανοχή σφάλματος* καθορίζεται εκ των προτέρων, τότε μπορούμε να σταματήσουμε όταν το $rB(\overline{\boldsymbol{x}})$ είναι μικρότερο από αυτήν την ποσότητα.

Δυστυχώς, δεν υπάρχει τέτοια εγγύηση για το μέγιστο σφάλμα, που να μπορεί να δοθεί για προβλήματα μη κυρτού προγραμματισμού. Ωστόσο, το $rB(\overline{\boldsymbol{x}})$ εξακολουθεί να είναι πιθανό να υπερβεί το πραγματικό σφάλμα, όταν τα $\overline{\boldsymbol{x}}$ και $\boldsymbol{x}^*$ αντιστοιχούν σε *τοπικά μέγιστα* του $P(\boldsymbol{x}; r)$ και του αρχικού προβλήματος, αντίστοιχα.

Περίληψη της SUMT.

Αρχικοποίηση: Προσδιορίζουμε μια *εφικτή* αρχική ενδιάμεση λύση $\boldsymbol{x}^{(\mathbf{0})}$, που δεν είναι στο σύνορο της εφικτής περιοχής. Θέτουμε $k = 1$ και επιλέγουμε τις κατάλληλες αυστηρά θετικές τιμές για το αρχικό r και για $\theta < 1$ (δηλαδή, $r = 1$ και $\theta = 0.01$).

Επαναλήψεις: Ξεκινώντας από το $\boldsymbol{x}^{(\boldsymbol{k}-\mathbf{1})}$, εφαρμόζουμε τη διαδικασία αναζήτησης μέσω κλίσης, που περιγράφεται στην Εν.3.2 (ή άλλη παρόμοια μέθοδο), για να βρούμε ένα τοπικό μέγιστο $\boldsymbol{x}^{(\boldsymbol{k})}$ της παράστασης:

$$P(\boldsymbol{x}; r) = f(\boldsymbol{x}) - r\left[\sum_{i=1}^{m} \frac{1}{b_i - g_i(\boldsymbol{x})} + \sum_{j=1}^{n} \frac{1}{x_j}\right].$$

Κριτήριο τερματισμού: Αν η αλλαγή από το $\boldsymbol{x}^{(\boldsymbol{k}-\mathbf{1})}$ στο $\boldsymbol{x}^{(\boldsymbol{k})}$ είναι αμελητέα, τότε σταματάμε και χρησιμοποιούμε το $\boldsymbol{x}^{(\boldsymbol{k})}$ (ή προέκτασή του $\boldsymbol{x}^{(\mathbf{0})}, \boldsymbol{x}^{(\mathbf{1})}, \ldots, \boldsymbol{x}^{(\boldsymbol{k}-\mathbf{1})}, \boldsymbol{x}^{(\boldsymbol{k})}$), ως εκτίμηση ενός *τοπικού μέγιστου* του αρχικού προβλήματος. Σε αντίθετη περίπτωση, επαναφέρουμε $k = k + 1$ και $r = \theta r$ και εκτελούμε επόμενη επανάληψη.

Όταν δεν ικανοποιούνται οι υποθέσεις του κυρτού προγραμματισμού, ο αλγόριθμος θα πρέπει να επαναληφθεί αρκετές φορές ξεκινώντας από μια ποικιλία εφικτών αρχικών ενδιάμεσων λύσεων. Το καλύτερο *τοπικό μέγιστο*, που λαμβάνεται για το αρχικό πρόβλημα θα πρέπει να χρησιμοποιηθεί ως η καλύτερη διαθέσιμη προσέγγιση του ολικού μέγιστου.

Τέλος, σημειώνουμε ότι η SUMT μπορεί, επίσης, να επεκταθεί για να συμπεριλάβει περιορισμούς *ισότητας* $g_i(\boldsymbol{x}) = b_i$. Ένας συνήθης τρόπος είναι ο ακόλουθος. Σε κάθε περιορισμό ισότητας,

η ποσότητα $\frac{-[b_i - g_i(x)]^2}{\sqrt{r}}$ αντικαθιστά την $\frac{-r}{b_i - g_i(x)}$

στην έκφραση του $P(\boldsymbol{x}; r)$, που δίνεται παραπάνω και στη συνέχεια χρησιμοποιείται η ίδια διαδικασία. Ο αριθμητής $-[b_i - g_i(\boldsymbol{x})]^2$ επιβάλλει μεγάλη ποινή, για ουσιώδεις αποκλίσεις από την ικανοποίηση των περιορισμών ισότητας και στη συνέχεια ο παρονομαστής αυξάνει δραματικά αυτή την ποινή καθώς το r μειώνεται πολύ, αναγκάζοντας έτσι την ακολουθία των ενδιάμεσων λύσεων να συγκλίνουν προς ένα σημείο, που ικανοποιεί τον περιορισμό.

Η SUMT έχει χρησιμοποιηθεί ευρέως λόγω της απλότητας και της ευελιξίας της. Ωστόσο, οι αριθμητικοί αναλυτές έχουν διαπιστώσει ότι είναι σχετικά επιρρεπής σε *αριθμητική αστάθεια*, οπότε συνιστάται μεγάλη προσοχή.

Παράδειγμα. Για να παρουσιάσουμε την SUMT, θεωρούμε το ακόλουθο πρόβλημα δύο μεταβλητών:

Μεγιστοποίησε: $f(\boldsymbol{x}) = x_1 x_2$,

υπό τους περιορισμούς:

$x_1^2 + x_2 \leq 3$

και

$x_1 \geq 0, x_2 \geq 0.$

Παρόλο που η $g_1(\boldsymbol{x}) = x_1^2 + x_2$ είναι κυρτή (διότι αποτελείται από κυρτούς όρους), έχουμε πρόβλημα *μη κυρτού* προγραμματισμού, επειδή η $f(\boldsymbol{x}) = x_1x_2$ *δεν* είναι κοίλη.

Για την αρχικοποίηση, η $(x_1, x_2) = (1, 1)$ είναι μια προφανής εφικτή λύση, που δεν είναι στο σύνορο της εφικτής περιοχής και έτσι μπορούμε να θέσουμε $\boldsymbol{x}^{(\mathbf{0})} = (1, 1)$. Λογικές επιλογές για το r και το θ είναι $r = 1$ και $\theta = 0.01$.

Για κάθε επανάληψη,

$$P(\boldsymbol{x}; r) = x_1x_2 - r(\tfrac{1}{3 - x_1^2 - x_2} + \tfrac{1}{x_1} + \tfrac{1}{x_2}).$$

Με $r = 1$, εφαρμόζοντας την διαδικασία αναζήτησης μέσω κλίσης, ξεκινώντας από το $(1, 1)$, για να μεγιστοποιηθεί η έκφραση αυτή οδηγούμαστε τελικά στο $\boldsymbol{x}^{(\mathbf{1})} = (0.90, 1.36)$. Επαναφέρουμε $r = 0.01$ και κάνουμε επανεκκίνηση της διαδικασίας αναζήτησης μέσω κλίσης από το $(0.90, 1.36)$, στη συνέχεια οδηγούμαστε στο $x^{(2)} = (0.983, 1.933)$. Μία ακόμα επανάληψη με $r = 0.01(0.01) = 0.0001$ οδηγεί από το $\boldsymbol{x}^{(\mathbf{2})}$ στο $\boldsymbol{x}^{(\mathbf{3})} = (0.998, 1.994)$. Αυτή η αλληλουχία των σημείων, τα οποία συνοψίζονται στον Πίνακα 4.5.Ι, σαφώς συγκλίνει στο $(1, 2)$. Εφαρμόζοντας τις συνθήκες ΚΚΤ σε αυτή τη λύση επαληθεύουμε ότι πληροί πράγματι την απαραίτητη προϋπόθεση για τη βελτιστότητα. Η γραφική ανάλυση καταδεικνύει ότι $(x_1, x_2) = (1, 2)$ είναι, πράγματι, το ολικό μέγιστο.

Για το πρόβλημα αυτό, δεν υπάρχουν άλλα τοπικά μέγιστα, εκτός από το $(x_1, x_2) = (1, 2)$. Έτσι, η επανεφαρμογή της SUMT, για διάφορες εφικτές αρχικές ενδιάμεσες λύσεις, οδηγεί πάντα στην ίδια λύση.

K	**r**	$\boldsymbol{x}_1^{(\boldsymbol{k})}$	$\boldsymbol{x}_2^{(\boldsymbol{k})}$
0		1	1
1	1	0.90	1.36
2	10^{-2}	0.983	1.933
3	10^{-4}	0.998	1.994
		↓	↓
		1	2

Πίνακας 4.5.Ι Παρουσίαση της SUMT.

Επίλογος

Συμπεράσματα

Κάποια πρακτικά προβλήματα βελτιστοποίησης περιλαμβάνουν συχνά *μη γραμμική* συμπεριφορά, η οποία πρέπει να ληφθεί υπ’ όψη. Μερικές φορές είναι δυνατό να *επαναδιατυπώσουμε* αυτά τα μη γραμμικά προβλήματα, για να τα εντάξουμε σε μια μορφή γραμμικού προγραμματισμού, όπως μπορεί να γίνει για τα προβλήματα *διαχωρίσιμου* προγραμματισμού. Ωστόσο, είναι συχνά απαραίτητη η χρήση μιας μορφοποίησης *μη γραμμικού προγραμματισμού*.

Σε αντίθεση με την περίπτωση της μεθόδου Simplex για γραμμικό προγραμματισμό, δεν υπάρχει αποδοτικός αλγόριθμος για όλες τις περιπτώσεις, που να μπορεί να χρησιμοποιηθεί, για να λύσει όλα τα προβλήματα μη γραμμικού προγραμματισμού. Πράγματι, κάποια από αυτά τα προβλήματα δεν μπορούν να επιλυθούν με πολύ ικανοποιητικό τρόπο, με οποιαδήποτε μέθοδο. Ωστόσο, έχει σημειωθεί σημαντική πρόοδος, για ορισμένες σημαντικές κατηγορίες προβλημάτων, συμπεριλαμβανομένων των προβλημάτων *τετραγωνικού προγραμματισμού*, *κυρτού προγραμματισμού*, καθώς και ορισμένων ειδικών τύπων προβλημάτων *μη κυρτού προγραμματισμού*. Ποικιλία από αλγόριθμους, που έχουν συχνά καλή συμπεριφορά, είναι διαθέσιμοι για αυτές τις περιπτώσεις. Μερικοί από αυτούς τους αλγόριθμους ενσωματώνουν εξαιρετικά αποτελεσματικές διαδικασίες για *βελτιστοποίηση χωρίς περιορισμούς*, για ένα τμήμα κάθε επανάληψης και μερικοί χρησιμοποιούν μία διαδοχή από γραμμικές ή τετραγωνικές προσεγγίσεις του αρχικού προβλήματος.

Τα τελευταία χρόνια, έχει δοθεί ισχυρή έμφαση στην ανάπτυξη υψηλής ποιότητας αξιόπιστων πακέτων λογισμικού, για γενική χρήση στην εφαρμογή των καλύτερων από αυτούς τους αλγορίθμους. Για παράδειγμα, έχουν αναπτυχθεί σημαντικά πακέτα λογισμικού, όπως το MINOS, στο Εργαστήριο Βελτιστοποίησης Συστημάτων, στο Πανεπιστήμιο του Stanford. Αυτά τα πακέτα χρησιμοποιούνται ευρέως και αλλού, για την επίλυση πολλών από τα είδη προβλημάτων, που συζητήθηκαν σε αυτό το κεφάλαιο (καθώς και για προβλήματα γραμμικού προγραμματισμού). Οι πάγιες βελτιώσεις, που πραγματοποιούνται, τόσο στις αλγοριθμικές τεχνικές, όσο και στο λογισμικό, φέρνουν μερικά πιο μεγάλα προβλήματα στο εύρος της υπολογιστικής δυνατότητας.

Η έρευνα στο μη γραμμικό προγραμματισμό παραμένει πολύ δραστήρια.

Βιβλιογραφία

1. Κολέτσος Ιωάννης – Στογιάννης Δημήτρης, «Εισαγωγή στην Επιχειρησιακή Έρευνα», Εκδόσεις Συμεών
2. Μπακόπουλος Α. – Χρυσοβέργης Ι., «Αριθμητικές μέθοδοι Μερικών Διαφορικών Εξισώσεων, Πεπερασμένα στοιχεία και διαφορές», Εκδόσεις Συμεών (2003)
3. Μπακόπουλος Α. – Χρυσοβέργης Ι., «Εισαγωγή στην Αριθμητική Ανάλυση, με Βιβλιοθήκη Προγραμμάτων και Δισκέτα», Εκδόσεις Συμεών
4. Μισυρλής Ν., «Αριθμητική Ανάλυση, Μία αλγοριθμική προσέγγιση»
5. Μισυρλής Ν., «Δομές Δεδομένων με C» (2005)
6. Μισυρλής Ν., «Εισαγωγή στις Δομές Δεδομένων (Pascal)», Εκδόσεις Συμμετρία
7. Μισυρλής Ν., «Εισαγωγή στον Προγραμματισμό με την C» (2005)
8. Μισυρλής Ν., «Pascal και Turbo Pascal», Εκδόσεις Συμμετρία (1996)
9. Bradle B., «A friendly introduction to Numerical Analysis, with C and Matlab Materials on Website», Prentice Hall. Pearson education international
10. Henrici P., «Elements of Numerical Analysis», John Wiley & Sons Inc.
11. Hillier - Lieberman, «Introduction to Operations Research», McGraw-Hill, 7th edition
12. Rudin W., «Functional Analysis», McGraw – Hill, 2nd edition
13. Sachs E. W. – Tichatschke, «System modeling and Optimization XX», Kluwer Academic Publisers
14. Sauer T., «Numerical Analysis», Pearson
15. Taha H. A., «Operations Research, an introduction», Prentice Hall, 8th edition (2003)
16. Winston W. L., «Operations Research, Applications and Algorithms», Prentice Hall, 4th edition