ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ
ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ
ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΑΠΟΦΑΣΕΩΝ

Ολοκληρωμένη Πολυκριτηριακή Μεθοδολογία & Πληροφοριακό Σύστημα Διαχείρισης Χαρτοφυλακίων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ελισσαίος Β. Σαρμάς

Επιβλέπων: Ιωάννης Ψαρράς
Καθηγητής Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2020

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ ΣΧΟΛΗ
ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ
ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ
ΑΠΟΦΑΣΕΩΝ

Ολοκληρωμένη Πολυκριτηριακή Μεθοδολογία & Πληροφοριακό Σύστημα Διαχείρισης Χαρτοφυλακίων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ελισσαίος Β. Σαρμάς

Επιβλέπων: Ιωάννης Ψαρράς
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την XX Φεβρουαρίου 2020.

....	....	....
Ψαρράς Ι.	Ασκούνης Δ.	Δούκας Χ.
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Αν. Καθηγητής Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2020

....

Ελισσαίος Β. Σαρμάς

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Πρόλογος

Θα ήθελα να αφιερώσω αυτή την διπλωματική σε όλους τους ανθρώπους που με βοήθησαν είτε άμεσα, είτε έμμεσα σε όλη την διάρκεια των φοιτητικών μου χρόνων.

Πρώτα από όλα, θα ήθελα να ευχαριστήσω τον Υπεύθυνο Καθηγητή της διπλωματικής μου κ. Πάνο Ξυδώνα για την αμέριστη υποστήριξή του, τις ιδέες που μου προσέφερε κάθε φορά που δεν ήξερα πως να συνεχίσω και την συνεχή καθοδήγησή του τους τελευταίους μήνες.

Θα ήθελα να ευχαριστήσω τον Καθηγητή και Διευθυντή του Εργαστηρίου κ. Ιωάννη Ψαρρά και τον Καθηγητή κ. Χάρη Δούκα, οι οποίοι από την πρώτη στιγμή μού έδειξαν εμπιστοσύνη, για την καθοδήγηση τους και την άριστη συνεργασία που έχουμε, σε όλο το χρονικό διάστημα της ενασχόλησης μου στις δραστηριότητες του Εργαστηρίου Συστημάτων Αποφάσεων και Διοίκησης.

Κατά την διάρκεια της πενταετούς φοίτησης μου στο ΕΜΠ είχα την τύχη να γνωρίσω πολλούς ανθρώπους που με βοήθησαν ο καθένας με τον δικό του τρόπο. Κάθε ένας από αυτούς με έχει επηρεάσει ξεχωριστά και με έχει βοηθήσει να βελτιωθώ.

Τέλος, από αυτή την λίστα δεν θα μπορούσε να λείπει η οικογένεια μου, η οποία με στήριζει σε κάθε βήμα της ζωής μου, σε κάθε εμπόδιο και σε κάθε δυσκολία.

Ελισσαίος Σαρμάς,

Αθήνα, Φεβρουάριος 2020

Περίληψη

Αντικείμενο της παρούσας Διπλωματικής Εργασίας αποτελεί η ανάπτυξη μιας ολοκληρωμένης μεθοδολογίας για την κατασκευή και διαχείριση χαρτοφυλακίου μετοχικών τίτλων. Η μεθοδολογία στηρίζεται σε δύο βασικούς άξονες: Ο πρώτος άξονας αφορά στο πρόβλημα της επιλογής των μετοχικών τίτλων, ενώ ο δεύτερος αφορά στο πρόβλημα της βελτιστοποίησης του επιλεχθέντος χαρτοφυλακίου μετοχών. Η προτεινόμενη μεθοδολογία αφορά μία συνδυαστική προσέγγιση των παραπάνω προβλημάτων, με αποτέλεσμα τη διαμόρφωση ενός πλήρους πλαισίου υποστήριξης αποφάσεων, το οποίο αποσκοπεί να συμπεριλάβει όλες τις παραμέτρους.

Ο πρώτος άξονας της μεθοδολογίας επικεντρώνεται στην επιλογή των μετοχικών τίτλων, οι οποίοι ενδεχομένως να συμπεριληφθούν στο τελικό χαρτοφυλάκιο. Για τη θεραπεία του συγκεκριμένου προβλήματος χρησιμοποιείται η Πολυκριτήρια Ανάλυση Αποφάσεων. Πιο συγκεκριμένα, ο αποφασίζων καλείται να επιλέξει τις αγορές και τους τομείς στους οποίους θα τοποθετηθεί. Στη συνέχεια, εφαρμόζονται τέσσερις βασικές μέθοδοι κατάταξης της Πολυκριτήριας Ανάλυσης Αποφάσεων στο σύνολο των επιλεχθέντων μετοχικών τίτλων: (α) η ELECTRE 3, (β) η PROMETHEE, (γ) η MAUT και (δ) η TOPSIS. Τέλος, ο πρώτος άξονας ολοκληρώνεται με την αθροιστική κατάταξη των μετοχικών τίτλων, βασισμένη στα επιμέρους αποτελέσματα κάθε μεθόδου.

Ο δεύτερος άξονας εστιάζει στη μοντελοποίηση μιας σειράς μεθοδολογιών για την επίλυση του προβλήματος της βελτιστοποίησης μετοχικού χαρτοφυλακίου. Ο αποφασίζων καλείται να καθορίσει το πλήθος των μετοχικών τίτλων που θα επιλεχθούν από την κατάταξη του προηγουμένου σταδίου. Εν συνεχεία, προτείνεται μια σειρά επιμέρους μεθοδολογιών για τη βελτιστοποίηση του χαρτοφυλακίου όπως: (α) η κλασσική μεθοδολογία μέσου-διακύμανσης, η οποία εμπλουτίζεται με μια πλήρη σειρά περιορισμών πολιτικής, (β) η μεθοδολογία του Προγραμματισμού Στόχων (γ) η μεθοδολογία πολυστοχικής βελτιστοποίησης, η οποία συμπεριλαμβάνει τη ροή της πολυκριτήριας μεθόδου PROMETHEE (δ) η μεθοδολογία που αφορά στη χρήση γενετικών αλγορίθμων για τη βελτιστοποίηση του χαρτοφυλακίου.

Στα πλαίσια της Διπλωματικής Εργασίας αναπτύχθηκε ένα ολοκληρωμένο πληροφοριακό σύστημα, το οποίο ενσωματώνει μερικές από τις σημαντικότερες μεθόδους της Πολυκριτήριας Ανάλυσης Αποφάσεων. Το πληροφοριακό σύστημα υλοποιήθηκε στη γλώσσα προγραμματισμού Python 3 και εξήχθη στον Παγκόσμιο Ιστό ως Διαδικτυακή Εφαρμογή μέσω του πλαισίου λογισμικού Django. Το πληροφοριακό σύστημα προσφέρει φιλική διεπαφή χρήστη και υλοποιεί αποδοτικά τις παραπάνω μεθόδους, παρέχοντας αναλυτική επίλυση πολυκριτηριακών προβλημάτων. Επιπλέον, ο δεύτερος άξονας της μεθοδολογίας αναπτύχθηκε, επίσης, στη γλώσσα προγραμματισμού Python 3 όπου υλοποιήθηκαν η οπτικοποίηση των δεδομένων, η στατιστική μελέτη των χρηματοοικονομικών δεικτών, καθώς και οι προαναφερθείσες τεχνικές βελτιστοποίησης χαρτοφυλακίου.

Η προτεινόμενη μεθοδολογία εφαρμόστηκε σε τέσσερα από τα μεγαλύτερα

διεθνή Χρηματιστήρια (NASDAQ, NYSE, Paris, Tokyo), σε τρεις από τους πιο αναπτυγμένους βιομηχανικούς τομείς (Τεχνολογικός, Ενεργειακός, Χρηματοοικονομικός), σε ένα σύνολο πάνω απο 2000 μετοχικών τίτλων. Τα δεδομένα για την εφαρμογή της μεθοδολογίας αντλήθηκαν από τις βάσεις δεδομένων των Yahoo Finance και Investing, ενώ η χρονική διάρκεια της ανάλυσης των δεδομένων ορίστηκε στα 3.5 έτη (Ιανουάριος 2016 - Ιούνιος 2019).

Λέξεις Κλειδιά: Πολυκριτήρια ανάλυση αποφάσεων, Χρηματοοικονομική μηχανική, Θεωρία βελτιστοποίησης χαρτοφυλακίου, Μετοχικοί τίτλοι, Σύγχρονη θεωρία χαρτοφυλακίου, Προγραμματισμός στόχων, Γραμμικός προγραμματισμός, Γενετικοί αλγόριθμοι, Πληροφοριακά συστήματα πολυκριτήριας ανάλυσης, Συστήματα υποστήριξης αποφάσεων

Abstract

The main object of this Thesis Project is the development of an integrated methodology for security portfolio management. The methodology consists of two basic phases: The first phase of the process involves the problem of security selection, while the second phase involves the problem of security portfolio optimisation. The proposed methodology includes an alternative combinatorial approach of the above problems, resulting in the configuration of a consistent decision support framework.

The first phase focuses on the selection of the securities which will potentially be included to the portfolio. The solution of this problem is approached with the Multiple-criteria Decision Analysis (MCDA). More specifically, initially the decision maker (DM) selects the market and the industrial sectors that he wishes to invest in. Subsequently, four MCDA ranking methods are applied to the pool of the selected securities: ELECTRE 3, PROMETHEE, MAUT and TOPSIS. Finally, the first phase includes the cumulative ranking of the securities, based on the individual ranking of each method.

The second phase focuses on the problem of portfolio optimisation. The DM should set the number of securities that will be selected from the cumulative ranking of the previous phase. In this point, four individual methodologies are proposed for portfolio optimisation: (a) the classic mean-variance methodology, equipped with a complete series of policy constraints, (b) the goal programming methodology, (c) the multiobjective programming methodology which includes the optimisation of PROMETHEE net flow and (d) the genetic algorithm methodology for portfolio optimisation.

As part of the Thesis project, an integrated information system was developed which includes some of the most important MCDA methods. The information system was developed in Python 3 programming language and was deployed as a Web Application with Django Web Framework. The information system offers a friendly Graphical User Interface (GUI) and efficiently implements a series of MCDA methods, exporting extensive solutions for a wide range of MCDA problems.

The validity of the proposed methodology is verified through an illustrative experimental application on four major international stock exchanges (NYSE, NASDAC, Paris, Tokyo) and 3 industrial sectors (Technological, Energy, Financial), including a pool of 2000 equities. The input data of the application were drawn from Yahoo Finance and Investing databases and the time horizon of the experimental application was set to 3.5 years (January 1, 2016 - June 31, 2019)

Keywords: Multicriteria decision analysis; Financial engineering; Portfolio optimisation theory; Security; Modern portfolio theory; Goal programming; Linear programming; Genetic Algorithms; Multicriteria decision support information systems; Decision support systems

Contents

List of Figures

List of Tables

Ευρεία Περίληψη

Κεφάλαιο 1: Εισαγωγή

Στη σημερινή εποχή, ένα από τα σημαντικότερα προβλήματα του χρηματοπιστωτικού τομέα είναι η δημιουργία και η διαχείριση ενός αποδοτικού χαρτοφυλακίου επενδύσεων στη βάση ενός πολύπλοκου περιβάλλοντος το οποίο χαρακτηρίζεται από ραγδαία αύξηση του ανταγωνισμού και σαρωτικές οικονομικές μεταβολές σε εθνικό και διεθνές επίπεδο. Γενικά, το χαρτοφυλάκιο επενδύσεων είναι ένα σύνολο περιουσιακών στοιχείων που αποκτήθηκαν με στόχο τη δημιουργία κέρδους για τον επενδυτή.

Μέχρι τη δεκαετία του 1950, η έννοια των χαρτοφυλακίων ήταν τελείως διαφορετική. Η επένδυση σε μετοχές αποτελούσε μια τυχαία διαδικασία καθώς δεν υπήρχαν επαρκή οικονομικά στοιχεία, ενώ ελάχιστοι άνθρωποι είχαν συνειδητοποιήσει τη σημασία της διαχείρισης των επενδύσεων. Οι επενδυτές επικεντρώνονταν συνήθως στις ευκαιρίες που προσφέρει κάθε μετοχή και όχι στη σχέση κέρδους-κινδύνου.

Η παραπάνω κατάσταση άλλαξε ριζικά από το 1952, όταν ο βραβευμένος με Νόμπελ H. Markowitz δημοσίευσε την ερευνητική του εργασία με τίτλο "Portfolio Selection", όπου εισήγαγε τη μαθηματική σχέση μεταξύ κέρδους και ρίσκου. Σύμφωνα με το μοντέλο μέσου-διακύμανσης του Markowitz, ένας συνδυασμός διαφορετικών ειδών μετοχών επιφέρει μικρότερο ρίσκο από μία μόνο μετοχή. Στη συνέχεια, οι επενδυτές ξεκίνησαν να δημιουργούν χαρτοφυλάκια που ευνοούσαν συγκεκριμένους επενδυτικούς τύπους και προτιμήσεις, χρησιμοποιώντας το μοντέλο μέσου-διακύμανσης ή άλλα μοντέλα που προσπάθησαν να το επεκτείνουν. Ως εκ τούτου, σήμερα η διαδικασία δημιουργίας και διαχείρισης χαρτοφυλακίων μετοχών έχει αναπτυχθεί και καλλιεργηθεί σημαντικά.

Ωστόσο, είναι γνωστό ότι η παγκόσμια οικονομία έχει ιστορικά αναταραχθεί από έντονες διακυμάνσεις, καθιστώντας τις μετοχές μία από τις πιο ευάλωτες αγορές. Τα χαρτοφυλάκια μετοχών είναι η πιο επικίνδυνη τοποθέτηση της αγοράς για δύο κύριους λόγους, σύμφωνα με τους Xidonas et al. (2010). Πρώτον, δεν υπάρχει δυνατότητα απαλοιφής μέρους του κινδύνου, επενδύοντας σε τίτλους σταθερής απόδοσης και σε παράγωγα προϊόντα. Δεύτερον, η διαδικασία διαχείρισης χαρτοφυλακίου μετοχών είναι εξαιρετικά δύσκολη λόγω της ύπαρξης μεγάλου αριθμού μετοχών που διαπραγματεύονται σε χρηματιστήρια. Το γεγονός αυτό καθιστά απαραίτητη τη διερεύνηση χιλιάδων τίτλων, οι οποίοι διατίθενται ως επενδυτικές επιλογές.

Η διαχείριση χαρτοφυλακίου είναι ένα πολύ περίπλοκο πρόβλημα, καθώς εδράζεται σε τρία διαφορετικά επίπεδα λήψης αποφάσεων: (i) επιλογή μετοχικών τίτλων που ενσωματώνουν τις καλύτερες επενδυτικές προοπτικές, (ii) διανομή του διαθέσιμου κεφαλαίου για την επίτευξη βέλτιστης σύνθεσης χαρτοφυλακίου και (iii) συγκριτική αξιολόγηση των κατασκευασμένων χαρτοφυλακίων.

Πέραν τούτου, το πρόβλημα της διαχείρισης χαρτοφυλακίου μετοχών συνδέεται με τρεις άλλες θεμελιώδεις παραμέτρους που επηρεάζουν κάθε διαδικασία λήψης αποφάσεων: (i) αβεβαιότητα, (ii) ύπαρξη πολλαπλών κριτηρίων και (iii) το προφίλ και τις προτιμήσεις του αποφασίζοντα.

Στόχος και αντικείμενο της διπλωματικής

Σήμερα, η ανάγκη για ανάπτυξη ολοκληρωμένων μεθοδολογικών πλαισίων και συστημάτων υποστήριξης αποφάσεων είναι ισχυρότερη από ποτέ. Τα πλαίσια αυτά πρέπει να ενσωματώνουν όλα τα κριτήρια και τις αλληλεπιδράσεις μεταξύ τους, καθώς και την αβεβαιότητα της χρηματοπιστωτικής αγοράς και τα διαφορετικά χαρακτηριστικά και τις ανάγκες των ενδιαφερομένων.

Αφετηρία αυτής της ερευνητικής προσπάθειας είναι η κλασική θεωρία μέσου - διακύμανσης του Markowitz. Αυτή η προσέγγιση είναι πολύ χρήσιμη αλλά δεν επαρκεί για την αποτελεσματική αντιμετώπιση του προβλήματος της διαχείρισης χαρτοφυλακίων μετοχών. Ως εκ τούτου, το προτεινόμενο μεθοδολογικό πλαίσιο επιδιώκει να απαλοίψει τις αδυναμίες του μοντέλου μέσου - διακύμανσης και να ξεπεράσει τις υπάρχουσες δυσκολίες υπολογισμού.

Το αντικείμενο της διπλωματικής εργασίας είναι η ανάπτυξη μιας ολοκληρωμένης μεθοδολογίας υποστήριξης αποφάσεων για τη διαχείριση χαρτοφυλακίων μετοχών, στο πλαίσιο της έντονης μεταβλητότητας και της αυξανόμενης αβεβαιότητας του σύγχρονου οικονομικού περιβάλλοντος.

Ο στόχος της διπλωματικής εργασίας είναι ο εντοπισμός όλων των παραμέτρων του προβλήματος, η εκτεταμένη ανάλυση των αλληλεπιδράσεων μεταξύ τους και τέλος η διαμόρφωση ενός διαφανούς και συνεπούς πλαισίου υποστήριξης αποφάσεων.

Συνεισφορά και αξία της διπλωματικής

Γενικά, η διπλωματική συμβάλλει στην επιστημονική κοινότητα ένα ολοκληρωμένο μεθοδολογικό πλαίσιο διαχείρισης χαρτοφυλακίου μετοχών, καθώς και ένα σύγχρονο σύστημα λήψης αποφάσεων. Επιπρόσθετα, κάθε μεμονωμένο στάδιο της μεθοδολογίας θα μπορούσε να εφαρμοστεί με επιτυχία, ακόμη και ξεχωριστά από το σύνολο του μεθοδολογικού πλαισίου. Συγκεκριμένα, η συμβολή της εργασίας συνοψίζεται στα παρακάτω σημεία:

Η διπλωματική συμβάλλει στη λεπτομερή επισκόπηση των υφιστάμενων γνώσεων στον τομέα της διαχείρισης χαρτοφυλακίων, παρουσιάζοντας τις βασικές έννοιες της σύγχρονης θεωρίας χαρτοφυλακίου. Επιπλέον, επιχειρείται μια λεπτομερής περιγραφή του τομέα της πολυκριτηριακής ανάλυσης αποφάσεων (MCDA).

Αναπτύσσεται ένα ολοκληρωμένο μεθοδολογικό πλαίσιο το οποίο φιλοδοξεί να συμπεριλάβει όλη τη διαδικασία διαχείρισης χαρτοφυλακίου. Το μεθοδολογικό πλαίσιο αποτελείται από δύο κύριες φάσεις: α) τη φάση της επιλογής χαρτοφυλακίου και β) τη φάση της βελτιστοποίησης του χαρτοφυλακίου.

Η πρώτη φάση της διαδικασίας βασίζεται σε μια πολυκριτηριακή μεθοδολογία λήψης αποφάσεων για την επιλογή των κατάλληλων μετοχικών τίτλων. Η διαδικασία αυτή βασίζεται σε τέσσερις θεμελιώδεις μεθόδους κατάταξης οι οποίες συνδυάζονται για να υποστηρίξουν την επιλογή των καλύτερων μετοχικών τίτλων.

Στο πλαίσιο της προτεινόμενης μεθοδολογίας αναπτύσσεται ένα ολοκληρωμένο υποσύστημα υπολογισμού χρηματοοικονομικών δεικτών. Αυτό το υποσύστημα περιλαμβάνει ορισμένα χρήσιμα εργαλεία εικονικοποίησης και υπολογισμού στατιστικών δεικτών.

Η δεύτερη φάση της μεθοδολογίας στηρίζεται σε συνεχείς μαθηματικές μεθόδους βελτιστοποίησης. Επομένως, έχουν αναπτυχθεί ορισμένα μοντέλα βελτιστοποίησης χαρτοφυλακίου όπως: τροποποίηση του μοντέλου μέσου-διακύμανσης, προσέγγιση με γενετικό αλγόριθμο και μεθοδολογία προγραμματισμού στόχων.

Τέλος, στο πλαίσιο αυτού του έργου αναπτύσσεται ένα πλήρες πληροφοριακό σύστημα για να υποστηριχθεί ολόκληρη η διαδικασία. Επιπλέον, το υποσύστημα πολυκριτήριας ανάλυσης αποφάσεων αναπτύχθηκε ως εφαρμογή web που εφαρμόζει μια σειρά μεθόδων πολλαπλών κριτηρίων με λεπτομερείς λύσεις βήμα προς βήμα.

Δομή της διπλωματικής

Η διπλωματική αποτελείται από έξι κεφάλαια και δύο παραρτήματα. Ακολουθεί μια σύντομη περιγραφή του περιεχομένου τους.

Κεφάλαιο 1

Στο 1ο κεφάλαιο της διπλωματικής γίνεται μια σύντομη εισαγωγή στο πρόβλημα, καθορίζοντας τα κύρια χαρακτηριστικά και το ιστορικό υπόβαθρο. Καθορίζονται ο στόχος της διπλωματικής καθώς και η συμβολή της στην επιστημονική κοινότητα. Τέλος, επιχειρείται μια περιγραφή της δομής της διπλωματικής.

Κεφάλαιο 2

Στο 2ο κεφάλαιο της εργασίας, πραγματοποιείται μια εισαγωγή στο πρόβλημα της διαχείρισης χαρτοφυλακίου. Το κεφάλαιο ξεκινά με τους θεμελιώδεις ορισμούς και την περιγραφή της έννοιας της διαφοροποίησης. Στη συνέχεια, παρουσιάζεται το πρόβλημα της βελτιστοποίησης του χαρτοφυλακίου με και χωρίς ανοιχτές πωλήσεις. Τέλος, εισάγεται η περίπτωση του ακίνδυνου χρεογράφου. Το κεφάλαιο περιέχει επίσης μερικές βασικές αποδείξεις των θεμελιωδών εξισώσεων.

Κεφάλαιο 3

Στο 3ο κεφάλαιο της διπλωματικής παρουσιάζονται οι συσχετιζόμενες μεθοδολογίες. Το κεφάλαιο χωρίζεται σε δύο μέρη. Στο πρώτο μέρος γίνεται μια εισαγωγή στις διακριτές μεθόδους ανάλυσης αποφάσεων πολλαπλών κριτηρίων, συμπεριλαμβανομένων μερικών βασικών ορισμών και μιας ιστορικής επισκόπησης. Στο δεύτερο μέρος, πραγματοποιείται μια εισαγωγή στον πολυκριτήριο μαθηματικό προγραμματισμό.

Κεφάλαιο 4

Στο 4ο κεφάλαιο της εργασίας παρουσιάζεται η προτεινόμενη μεθοδολογία. Αρχικά, γίνεται μια επισκόπηση της μεθοδολογίας μέσω επεξηγηματικών διαγραμμάτων. Εν συνεχεία, περιγράφονται λεπτομερώς οι δύο φάσεις της μεθοδολογίας. Η πρώτη φάση αφορά στην επιλογή χαρτοφυλακίου, συμπεριλαμβανομένου του ψευδοκώδικα των μεθόδων κατάταξης. Η δεύτερη φάση αφορά στη βελτιστοποίηση χαρτοφυλακίων υπό πολλαπλά κριτήρια, και περιλαμβάνει τα προτεινόμενα μοντέλα της διπλωματικής.

Κεφάλαιο 5

Στο 5ο κεφάλαιο της εργασίας, παρουσιάζουμε το πληροφοριακό σύστημα που υλοποιεί τη μεθοδολογία. Το κεφάλαιο εισάγει όλα τα εργαλεία και τις βιβλιοθήκες που χρησιμοποιήθηκαν. Επιπλέον, περιλαμβάνει τα βασικά διαγράμματα UML που περιγράφουν το πληροφοριακό σύστημα. Στο δεύτερο μέρος του κεφαλαίου παρουσιάζεται μια σύντομη παρουσίαση της εφαρμογής. Τέλος, στο τελευταίο μέρος του κεφαλαίου παρουσιάζεται ένα μέρος του πηγαίου κώδικα.

Κεφάλαιο 6

Στο 6ο κεφάλαιο της διπλωματικής παρουσιάζεται ένα μέρος των αποτελεσμάτων της πειραματικής εφαρμογής της προτεινόμενης μεθοδολογίας. Συγκεκριμένα, παρουσιάζονται τα αποτελέσματα κάθε βήματος της μεθοδολογίας για την αγορά της Νέας Υόρκης. Στο τέλος του κεφαλαίου, πραγματοποιείται η επαλήθευση της μεθοδολογίας σε out-of-sample δεδομένα.

Κεφάλαιο 7

Τέλος, στο 7ο κεφάλαιο της εργασίας διατυπώνονται τα συμπεράσματα του συνόλου του έργου και συζητούνται οι κύριες μελλοντικές προοπτικές.

Παράρτημα Α

Στο παράρτημα Α, παρουσιάζεται ο κύριος όγκος του πηγαίου κώδικα, συνοδευόμενος από βήμα προς βήμα αποτελέσματα και σύντομα σχόλια.

Παράρτημα Β

Στο παράρτημα Β παρουσιάζεται ο κύριος όγκος των αποτελεσμάτων της εφαρμογής της προτεινόμενης μεθοδολογίας. Τα αποτελέσματα που παρουσιάζονται σε αυτό το μέρος αναφέρονται σε τρεις από τους μεγαλύτερους βιομηχανικούς τομείς (τεχνολογικός, ενεργειακός και οικονομικός) και σε τέσσερα μεγάλα χρηματιστήρια (NYSE, NASDAQ κ.λπ.).

Κεφάλαιο 2: Το πρόβλημα της διαχείρισης χαρτοφυλακίου

Στο παρόν κεφάλαιο αναπτύσσεται το πρόβλημα της διαχείρισης μετοχικού χαρτοφυλακίου και ειδικότερα της μεθοδολογίας μέσου - διακύμανσης που αναπτύχθηκε από τον Harry Markowitz (1952, 1959). Το πρόβλημα της σύνθεσης του χαρτοφυλακίου εισήχθη ως ένα τετραγωνικό πρόβλημα μαθηματικού προγραμματισμού. Από τότε, πολλοί επιστήμονες προσπάθησαν να βελτιώσουν αυτή τη μεθοδολογία και να θεραπεύσουν τις αδυναμίες της, χρησιμοποιώντας ποικίλες τεχνικές βελτιστοποίησης και άλλες μεθόδους επιχειρησιακής έρευνας. Η παρουσίαση του μεθοδολογικού πλαισίου μέσου - διακύμανσης αναπτύσσεται σε τέσσερις ενότητες.

Στην πρώτη ενότητα γίνεται μια σύντομη επισκόπηση των βασικών εννοιών που αποτελούν το πρόβλημα. Στη δεύτερη ενότητα παρουσιάζεται η θεμελιώδης αρχή της διαφοροποίησης. Γίνεται μια εισαγωγή στίς δύο διαφορετικές συνιστώσες του κινδύνου (συστηματικός και μη συστηματικός κίνδυνος), αναλύοντας τους παράγοντες που καθιστούν αναγκαία την υιοθέτηση μιας στρατηγικής διαφοροποίησης. Στην τρίτη ενότητα πραγματοποιείται μια λεπτομερής περιγραφή του προβλήματος της βελτιστοποίησης του χαρτοφυλακίου. Εισάγονται οι έννοιες των αποτελεσματικών χαρτοφυλακίων και του αποτελεσματικών μετώπου. Αναλύεται τόσο η περίπτωση που επιτρέπονται οι ανοιχτές πωλήσεις όσο και η περίπτωση που δεν επιτρέπονται. Τέλος, εισάγεται η έννοια του ακίνδυνου χρεογράφου. Η ανάλυση χωρίζεται και πάλι ανάλογα με το καθεστώς ανοίχτών πωλήσεων, ενώ τέλος, αναπτύσσονται δύο διαφορετικές αποδείξεις για την περίπτωση ανοιχτών πωλήσεων.

Το κεφάλαιο αυτό οδήγησε στα ακόλουθα συμπεράσματα:

- Το μοντέλο μέσου - διακύμανσης βασίζεται σε ένα τετραγωνικό πρόβλημα μαθηματικού προγραμματισμού το οποίο περιλαμβάνει σημαντική υπολογιστική πολυπλοκότητα. Ειδικότερα, ο υπολογισμός του πίνακα συνδιακύμανσης καθίσταται πολύ δύσκολος σε περίπτωση που ο αριθμός των χρεογράφων είναι μεγάλος. Η πολυπλοκότητα του αλγορίθμου για είσοδο n μετοχικών τίτλων είναι της τάξεως $O(n^2)$, καθιστώντας έτσι το πρόβλημα μη γραμμικό.
- Το προτεινόμενο μοντέλο βασίζεται σε δύο κριτήρια (απόδοση και κίνδυνος), αποτυγχάνοντας έτσι να απεικονίσει ρεαλιστικά όλες τις παραμέτρους του προβλήματος. Αντίθετα, μια ολοκληρωμένη προσέγγιση απαιτεί την ενσωμάτωση όλων των παραμέτρων που επηρεάζουν την αγορά. Κατά συνέπεια, το πρόβλημα διαχείρισης χαρτοφυλακίου είναι ένα πρόβλημα πολλαπλών κριτηρίων, όπως και η πλειοψηφία των προβλημάτων λήψης αποφάσεων σήμερα.
- Η συμβατική προσέγγιση δεν λαμβάνει υπόψην το προφίλ του αποφασίζοντα. Οι προτιμήσεις του αποφασίζοντα έχουν τεράστια σημασία στη διαδικασία διαχείρισης χαρτοφυλακίου. Για παράδειγμα, ένας συντηρητικός επενδυτής πιθανώς θα ενδιαφερόταν για την ελαχιστοποίηση του κινδύνου, ενώ αντίθετα ένας επιθετικός επενδυτής θα προτιμούσε να μεγιστοποιήσει την απόδοση.

Εν κατακλείδι, το πρόβλημα της διαχείρισης χαρτοφυλακίου είναι ένα πρόβλημα πολλαπλών κριτηρίων καθώς περιλαμβάνει πολλούς παράγοντες. Κατά συνέπεια, η ανάγκη για ολοκληρωμένες μεθοδολογίες είναι επιτακτική προκειμένου να επιλυθούν τα προαναφερθέντα προβλήματα.

Κεφάλαιο 3: Επισκόπηση συναφών μεθοδολογιών

Η ανάλυση του προβλήματος διαχείρισης χαρτοφυλακίου σηματοδότησε την ανάγκη για νέα μεθοδολογικά πλαίσια υποστήριξης αποφάσεων, προκειμένου να υπερκεραστούν τα υπάρχοντα προβλήματα και να θεραπευθούν οι ανεπάρκειες του συμβατικού μοντέλου μέσου - διακύμανσης.

Στο τρίτο κεφάλαιο γίνεται μια σύντομη εισαγωγή στην ανάλυση αποφάσεων με πολλαπλά κριτήρια, καθώς είναι το καταλληλότερο μεθοδολογικό εργαλείο υποστήριξης της διαδικασίας λήψης αποφάσεων στο πρόβλημα της διαχείρισης χαρτοφυλακίου.

Στην πρώτη ενότητα παρουσιάζονται οι βασικές έννοιες αυτού του επιστημονικού πεδίου, καθώς και μια γενική μεθοδολογική επισκόπηση, αναλύοντας τις τέσσερις φάσεις υποστήριξης αποφάσεων. Επιπρόσθετα, γίνεται μια εισαγωγή στις διακριτές πολυκριτήριες μεθόδους υποστήριξης αποφάσεων, παρουσιάζοντας τους τρεις βασικούς τομείς (multiattribute utility theory, outranking relations theory, preference disaggregation approach).

Στη δεύτερη ενότητα αναπτύσσονται οι τεχνικές συνεχούς βελτιστοποίησης. Ξεκινώντας από τη θεμελιώδη έννοια του γραμμικού προγραμματισμού, στις επόμενες παραγράφους γίνεται μια εισαγωγή στον τετραγωνικό και τον ακέραιο προγραμματισμό. Στη συνέχεια παρουσιάζεται το μεθοδολογικό πλαίσιο των προβλημάτων προγραμματισμού στόχων, ακολουθούμενο από μια εισαγωγή σε γενετικούς αλγόριθμους.

Βασικά στοιχεία και μεθοδολογικό πλαίσιο

Το 1985, ο Roy, ένας από τους ιδρυτές της σύγχρονης θεωρίας πολυκριτηριακής ανάλυσης, παρουσίασε ένα γενικό μεθοδολογικό πλαίσιο για πολυδιάστατα προβλήματα λήψης αποφάσεων. Η διαδικασία ανάλυσης πολυκριτηριακών προβλημάτων περιλαμβάνει τέσσερα φάσεις:

Φάση 1: Αντικείμενο της απόφασης

Σε αυτή τη φάση, υπάρχουν δύο απαραίτητα βήματα: (a) Αυστηρός ορισμός του συνόλου A των εναλλακτικών λύσεων του προβλήματος και (b) προσδιορισμός της προβληματικής της απόφασης.

Το σύνολο A των εναλλακτικών λύσεων του προβλήματος μπορεί να είναι ένα συνεχές ή ένα διακριτό σύνολο. Στην περίπτωση ενός συνεχούς προβλήματος, το συνεχές σύνολο λύσεων ορίζεται από μαθηματικές εξισώσεις με τόσες διαστάσεις όσες το πλήθος των μεταβλητών απόφασης. Στην περίπτωση ενός διακριτού προβλήματος, το σύνολο των εφικτών λύσεων καθορίζεται με εξαντλητική απαρίθμηση των στοιχείων του.

Η προβληματική της απόφασης καθορίζει τον τρόπο με τον οποίο πρέπει να εξεταστούν οι εναλλακτικές λύσεις. Σύμφωνα με τον Roy, υπάρχουν τέσσερις κύριες κατηγορίες διακριτών προβλημάτων:

i. Τα προβλήματα επιλογής αφορούν την κατάσταση όπου ο αποφασίζων καλείται να επιλέξει τις καταλληλότερες εναλλακτικές λύσεις.

ii. Τα προβλήματα ταξινόμησης αφορούν την κατάσταση κατά την οποία οι εναλλακτικές λύσεις πρέπει να ταξινομηθούν σε προκαθορισμένες κατηγορίες.

iii. Τα προβλήματα κατάταξης αφορούν την κατάσταση κατά την οποία οι εναλλακτικές λύσεις πρέπει να καταταχθούν κατά φθίνουσα σειρά.

iv. Τα προβλήματα περιγραφής αναφέρονται στην κατάσταση όπου οι εναλλακτικές λύσεις περιγράφονται σύμφωνα με την επιδόσεις σε μεμονωμένα κριτήρια.

Φάση 2: Συνεπής οικογένεια κριτηρίων

Κάθε παράγοντας που επηρεάζει μια απόφαση θεωρείται κριτήριο. Τυπικά, ένα κριτήριο είναι μια μονότονη συνάρτηση f η οποία δηλώνει την προτίμηση του αποφασίζοντα, έτσι ώστε για οποιεσδήποτε δύο εναλλακτικές x_i, x_j να ισχύουν οι ακόλουθες εξισώσεις:

$$f(x_i) > f(x_j) \leftrightarrow x_i \succ x_j \tag{1}$$

$$f(x_i) = f(x_j) \leftrightarrow x_i \sim x_j \tag{2}$$

όπου ο συμβολισμός $x_i \succ x_j$ δηλώνει ότι η εναλλακτική x_i είναι προτιμότερη από τη x_j και ο συμβολισμός $x_i \sim x_j$ δηλώνει ότι υπάρχει αδιαφορία μεταξύ των δύο εναλλακτικών επιλογών.

Αυτή η διαδικασία έχει ως αποτέλεσμα τη διαμόρφωση μιας συνεπούς οικογένειας κριτηρίων. Ένα σύνολο κριτηρίων $F = \{f_1, \ldots, f_q\}$ διαμορφώνει μια συνεπή οικογένεια κριτηρίων, αν και μόνο εάν πληρούνται οι ακόλουθες ιδιότητες:

1. **Μονοτονία:** Ένα σύνολο κριτηρίων θεωρείται μονότονο εάν και μόνο εάν για κάθε δυο εναλλακτικές λύσεις x_i, x_j αν $f_k(x_i) > f_k(x_j)$ για οποιοδήποτε κριτήριο k και $f_l(x_i) = f_l(x_j)$ για οποιοδήποτε άλλο κριτήριο $l\ neqk$, συμπεραίνεται ότι το $x_i \succ x_j$.

2. **Επάρκεια:** Ένα σύνολο κριτηρίων θεωρείται ότι διαθέτει την ιδιότητα της επάρκειας εάν και μόνο εάν για δύο εναλλακτικές λύσεις x_i, x_j, αν $f_k(x_i) = f_k(x_j)$ για οποιοδήποτε κριτήριο k, συμπεραίνεται ότι $x_i \sim x_j$.

3. **Μη πλεονασμός:** Ένα σύνολο κριτηρίων θεωρείται ότι δεν είναι πλεονάζον εάν και μόνο εάν η κατάργηση οποιουδήποτε κριτηρίου οδηγεί σε παραβίαση των ιδιοτήτων της μονοτονίας ή της εξαντλητικότητας.

Φάση 3: Πρότυπο Συνολικής Αξιολόγησης

Το πρότυπο συνολικής αξιολόγησης ορίζεται ως η σύνθεση όλων των κριτηρίων, προκειμένου να αναλυθεί το πρόβλημα σύμφωνα με τα καθορισμένα προβλήματα. Το μοντέλο της συνολικής αξιολόγησης μπορεί να εφαρμοστεί για να προσδιοριστεί μια συνολική αξιολόγηση των εναλλακτικών λύσεων, να διερευνηθεί το σύνολο λύσεων (για συνεχή προβλήματα) και να εκτελεστούν οι συγκρίσεις μεταξύ όλων των ζευγών εναλλακτικών.

Φάση 4: Υποστήριξη λήψης αποφάσεων

Αυτή η φάση της διαδικασίας περιλαμβάνει όλες τις δραστηριότητες που βοηθούν τον αποφασίζοντα να κατανοήσει τα αποτελέσματα της εφαρμογής του μοντέλου. Ο ρόλος του συμβούλου είναι καθοριστικής σημασίας επειδή πρέπει να οργανώσει τις απαντήσεις με έναν κατανοητό τρόπο.

Συμπεράσματα

Η επισκόπηση των συναφών πολυκριτήριων μεθοδολογιών οδηγεί στα ακόλουθα συμπεράσματα:

- Έχουν αναπτυχθεί ποικίλες μέθοδοι υποστήριξης αποφάσεων με πολλαπλά κριτήρια, οι οποίες θα μπορούσαν να εφαρμοστούν σε ποικίλα διακριτά και συνεχή προβλήματα λήψης αποφάσεων. Αυτές οι τεχνικές είναι ικανές να αντιμετωπίζουν προβλήματα με πολλά αντικρουόμενα κριτήρια, γεγονός που τις καθιστά πολύ χρήσιμες στο πρόβλημα διαχείρισης χαρτοφυλακίου.
- Μερικές μέθοδοι πολλαπλών κριτηρίων (τετραγωνικός προγραμματισμός, σύνθετοι γενετικοί αλγόριθμοι κλπ.) έχουν μεγάλο υπολογιστικό φόρτο. Επομένως, σε περίπτωση που ο αριθμός των εναλλακτικών είναι μεγάλος, η επίλυση ορισμένων προβλημάτων είναι δύσκολη ή ακόμα και ανέφικτη, καθιστώντας απαραίτητη την εξεύρεση μιας προσεγγιστικής λύσης ή ενός συνόλου λύσεων.
- Ο επιστημονικός τομέας της διαχείρισης χαρτοφυλακίου έχει σημαντικό περιθώριο ανάπτυξης. Η ανάγκη ολοκληρωμένων μεθοδολογιών και συστημάτων υποστήριξης αποφάσεων είναι μεγάλη, προκειμένου να βελτιωθούν οι υπάρχουσες μεθοδολογίες.

Κεφάλαιο 4: Προτεινόμενη μεθοδολογία

Εισαγωγή

Όπως έχει ήδη αναφερθεί, ο σκοπός της διπλωματικής εργασίας είναι η ανάπτυξη ενός ολοκληρωμένου μεθοδολογικού πλαισίου για τη διαχείριση χαρτοφυλακίου. Η διαδικασία διαχείρισης χαρτοφυλακίου είναι ένα πολύ περίπλοκο πρόβλημα, καθώς αποτελείται από δύο στάδια που απαιτούν μια σειρά σημαντικών αποφάσεων. Η πρώτη φάση επικεντρώνεται στην επιλογή χαρτοφυλακίου, δηλαδή στην επιλογή των ισχυρότερων επενδυτικών ευκαιριών. Η δεύτερη φάση περιλαμβάνει τη βελτιστοποίηση του χαρτοφυλακίου, δηλαδή τον προσδιορισμό της αποδοτικότερης κατανομής του διαθέσιμου κεφαλαίου στα επιλεγμένα χρεόγραφα προκειμένου να μεγιστοποιηθεί η απόδοση.

Σε αυτό το κεφάλαιο, παρουσιάζεται η προτεινόμενη μεθοδολογία στο πλαίσιο της διπλωματικής. Ο απώτερος στόχος είναι η αποτελεσματική διαχείριση μετοχικών χαρτοφυλακίων, τα οποία αποτελούν μία από τις πιο επικίνδυνες επενδύσεις στην αγορά. Η προτεινόμενη μεθοδολογία φιλοδοξεί να συνδυάσει τις υπάρχουσες γνώσεις με ένα σύνολο θεωρητικών και πρακτικών καινοτομιών. Στην πρώτη φάση αναπτύσσονται τέσσερις μέθοδοι λήψης αποφάσεων πολλαπλών κριτηρίων για την ταξινόμηση των διαθέσιμων τίτλων και την ανίχνευση των καλύτερων επενδυτικών ευκαιριών. Μετά τη διαδικασία επιλογής χαρτοφυλακίου, υπολογίζονται οι σημαντικότεροι χρηματοοικονομικοί δείκτες βάσει ιστορικών δεδομένων και, εν συνεχεία, προτείνεται μια σειρά από μοντέλα βελτιστοποίησης χαρτοφυλακίου. Η βάση αυτών των μοντέλων είναι η κλασσική μέθοδος μέσου - διακύμανσης, η οποία παραμένει η βασική μέθοδος βελτιστοποίησης χαρτοφυλακίου για περισσότερα από 60 χρόνια. Τέλος, τα επιλεγμένα χαρτοφυλάκια αξιολογούνται και συγκρίνονται προκειμένου να επιλεχθεί το καταλληλότερο χαρτοφυλάκιο σύμφωνα με το προφίλ του αποφασίζοντος.

Επισκόπηση Μεθοδολογίας

Ένα εκτεταμένο διάγραμμα του προτεινόμενου μεθοδολογικού πλαισίου παρουσιάζεται στο διάγραμμα 1.

Φάση Ι: Επιλογή χαρτοφυλακίου

Η πρώτη φάση αφορά στο πρόβλημα της επιλογής χαρτοφυλακίου, δηλαδή την ανάπτυξη ενός συνόλου μετοχικών τίτλων οι οποίοι θεωρούνται επενδυτικές ευκαιρίες. Ο αποφασίζων καλείται να επιλέξει τον βιομηχανικό τομέα και την αγορά στην οποία επιθυμεί να επενδύσει, με συνέπεια τη δημιουργία ενός συνόλου μετοχικών τίτλων που αποτελούν τις εναλλακτικές λύσεις του προβλήματος. Το πρόβλημα της επιλογής χαρτοφυλακίου επιλύεται με την χρήση της ανάλυσης αποφάσεων με πολλαπλά κριτήρια (MCDA). Πιο συγκεκριμένα, στις εναλλακτικές λύσεις εφαρμόζονται τέσσερις πολυκριτήριες μέθοδοι, βάσει ποικίλων οικονομικών δεικτών οι οποίοι χρησιμεύουν ως

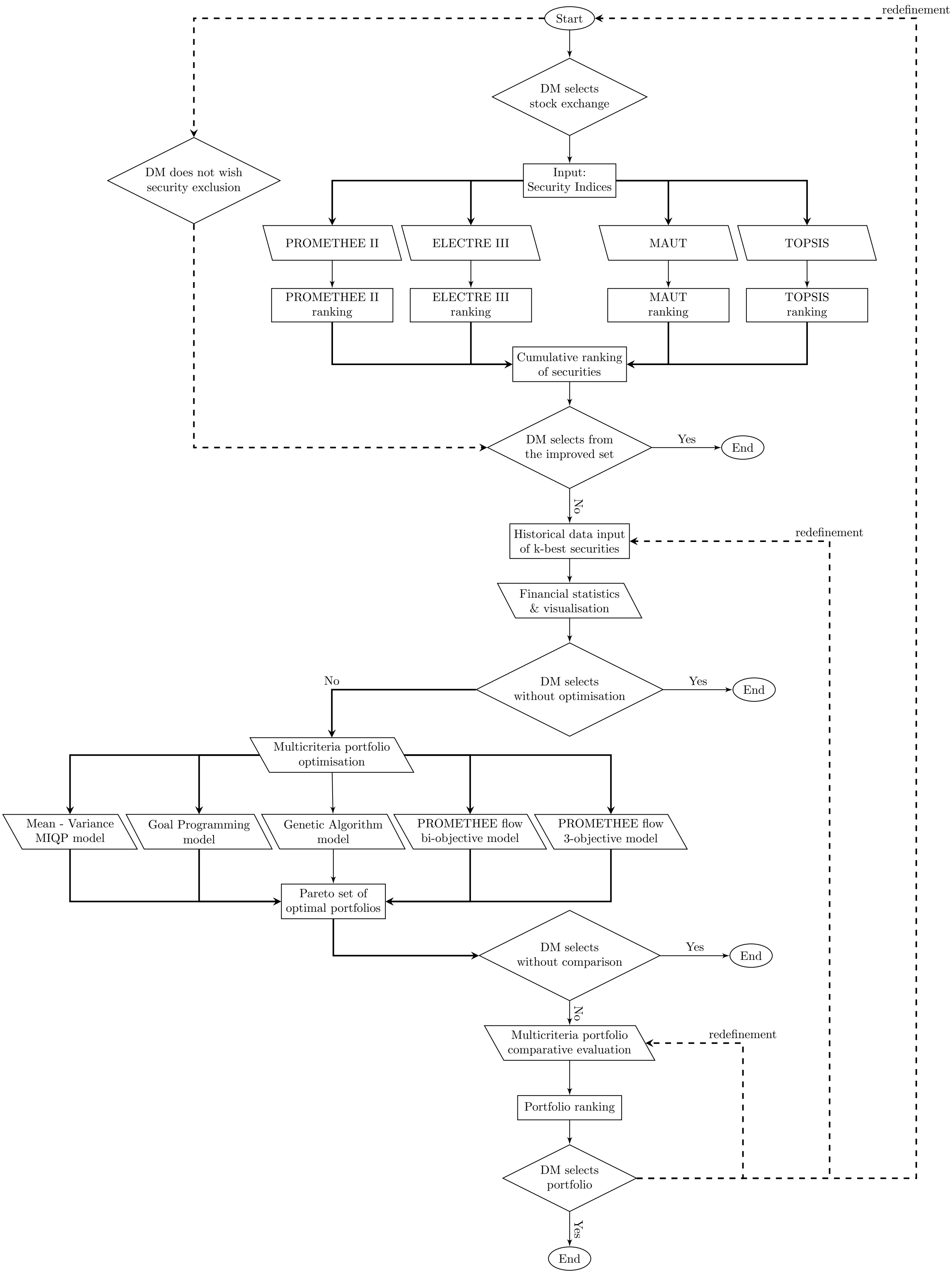


Figure 1: Εκτενής παρουσίαση μεθοδολογικού πλαισίου

κριτήρια του προβλήματος. Κάθε μέθοδος παρέχει μια κατάταξη των μετοχικών τίτλων, με συνέπεια η συνολική κατάταξη των τίτλων να υπολογίζεται ως ο σταθμισμένος μέσος όρος των τεσσάρων μεμονωμένων κατατάξεων. Μετά το πέρας τη διαδικασίας, ο αποφασίζων είτε επιλέγει το χαρτοφυλάκιο με τους καλύτερους μετοχικούς τίτλους όπως προέκυψαν από την παραπάνω διαδικασία, είτε επαναπροσδιορίζει το πρόβλημα σε περίπτωση που το αποτέλεσμα κριθεί μη ικανοποιητικό.

Φάση II: Βελτιστοποίηση χαρτοφυλακίου

Η δεύτερη φάση αφορά στο πρόβλημα της βελτιστοποίησης του χαρτοφυλακίου. Το αρχικό πρόβλημα βελτιστοποίησης διαμορφώθηκε ως πρόβλημα δύο κριτηρίων όπου η αναμενόμενη απόδοση του χαρτοφυλακίου πρέπει να μεγιστοποιηθεί, ενω ο κίνδυνος του χαρτοφυλακίου πρέπει να ελαχιστοποιηθεί (Markowitz 1952). Το μεθοδολογικό πλαίσιο της παρούσας εργασίας προτείνει μια σειρά από μοντέλα για την προσέγγιση του προβλήματος. Πρώτον, διατυπώνεται ένα εναλλακτικό μοντέλο μεικτού ακέραιου προγραμματισμού όπου επιβάλλονται επιπρόσθετοι ακέραιοι περιορισμοί για τον έλεγχο του συντελεστή στάθμισης κάθε χρεογράφου. Επιπρόσθετα, παρουσιάζεται μια διστοχική προσέγγιση, όπου μεγιστοποιείται η ροή της πολυκριτήριας μεθόδου PROMETHEE και ελαχιστοποιείται ο δείκτης beta του χαρτοφυλακίου. Τέλος, εισάγεται μια μεθοδολογία προγραμματισμού στόχων, καθώς και ένα μοντέλο γενετικού αλγορίθμου.

Στην τελική φάση επιχειρείται μια συγκριτική ανάλυση των χαρτοφυλακίων που παρήχθησαν. Τα χαρτοφυλάκια που προέκυψαν από τις παραπάνω μεθόδους βελτιστοποίησης συγκρίνονται ώστε να επιλεχθούν τα πιο κατάλληλα χαρτοφυλάκια σύμφωνα με το προφίλ του αποφασίζοντος.

Συμπεράσματα

Στο παρόν κεφάλαιο πραγματοποιήθηκε μια πλήρης παρουσίαση της προτεινόμενης μεθοδολογίας, η οποία φιλοδοξεί να ενσωματώσει όλες τις παραμέτρους του προβλήματος της διαχείρισης χαρτοφυλακίου. Η προτεινόμενη μεθοδολογία χωρίστηκε σε δύο φάσεις: (α) στην πρώτη φάση παρουσιάστηκε μια ολοκληρωμένη μεθοδολογία για το πρόβλημα της επιλογής χαρτοφυλακίου και (β) στη δεύτερη φάση παρουσιάστηκε μια σειρά εναλλακτικών μεθοδολογιών που στοχεύουν στην επίλυση του προβλήματος της βελτιστοποίησης χαρτοφυλακίου.

Για λόγους πληρότητας, είναι απαραίτητο να πραγματοποιηθεί μια σύντομη περιγραφή της τελικής φάσης του προβλήματος, η οποία περιλαμβάνει την επιλογή του καταλληλότερου χαρτοφυλακίου από το σύνολο αποτελεσματικών χαρτοφυλακίων. Σε αυτή τη φάση δεδομένου ενός pareto βέλτιστου συνόλου υποψήφιων χαρτοφυλακίων, το πρόβλημα έγκειται στον προσδιορισμό του καταλληλότερου χαρτοφυλακίου. Είναι προφανές ότι η πιο σημαντική παράμετρος του προβλήματος είναι το προφίλ του αποφασίζοντος, το οποίο καθορίζει τον τρόπο με τον οποίο θα πρέπει να υποστηριχθεί στη λήψη αποφάσεων.

Ωστόσο, σε περίπτωση που ο επενδυτής δεν έχει καταλήξει σε τελική απόφαση, αναπτύσσεται ένα μεθοδολογικό πλαίσιο για την υποστήριξη της λήψης απόφασεων. Το πρόβλημα της επιλογής του καταλληλότερου χαρτοφυλακίου από το σύνολο των βέλτιστων χαρτοφυλακίων μπορεί να λυθεί ως διακριτό πολυκριτήριο πρόβλημα, όπου οι εναλλακτικές λύσεις είναι όλα τα αποτελεσματικά χαρτοφυλάκια και τα κριτήρια μπορούν να καθοριστούν σε επικοινωνία με τον αποφασίζοντα. Ως εκ τούτου, το μεθοδολογικό πλαίσιο που χρησιμοποιήθηκε στην Φάση Ι για την επιλογή των καταλληλότερων χρεογράφων μπορεί να χρησιμοποιηθεί και σε αυτό το πρόβλημα για τον προσδιορισμό του καταλληλότερου χαρτοφυλακίου. Μετά την εφαρμογή των πολυκριτήριων μεθόδων επιτυγχάνεται η τελική κατάταξη των χαρτοφυλακίων και τέλος ο επενδυτής έχει τη δυνατότητα να επιλέξει το καταλληλότερο χαρτοφυλάκιο.

Συνοψίζοντας, το προτεινόμενο μεθοδολογικό πλαίσιο υποστηρίζει τον αποφασίζοντα με ένα ολοκληρωμένο μοντέλο, εξετάζοντας κάθε φάση της διαδικασίας. Επιπλέον, το πιο σημαντικό επίτευγμα είναι η ενσωμάτωση των προτιμήσεων του επενδυτή και η αλληλεπίδραση με τον αποφασίζοντα σε κάθε σημείο της διαδικασίας.

Κεφάλαιο 5: Πληροφοριακό σύστημα

Η παρουσίαση της προτεινόμενης μεθοδολογίας υπογράμμισε την ανάγκη για σύγχρονα πληροφοριακά συστήματα, τα οποία υλοποιούν τις προτεινόμενες μεθόδους. Στο πλαίσιο της παρούσας διπλωματικής σχεδιάστηκε και αναπτύχθηκε ένα πληροφοριακό σύστημα υποστήριξης αποφάσεων. Σκοπός του πληροφοριακού συστήματος είναι η αποτελεσματική εφαρμογή των αλγορίθμων που της προτεινόμενης μεθοδολογίας για την υποστήριξη της διαδικασίας λήψης αποφάσεων.

Το πληροφοριακό σύστημα αποτελείται από τέσσερα υποσυστήματα. Το πρώτο υποσύστημα περιλαμβάνει την εφαρμογή των πολυκριτήριων μεθόδων υποστήριξης αποφάσεων που χρησιμοποιούνται στην προτεινόμενη μεθοδολογία. Το δεύτερο υποσύστημα υποστηρίζει τον υπολογισμό των οικονομικών στατιστικών δεικτών. Το τρίτο υποσύστημα εφαρμόζει τις μεθόδους βελτιστοποίησης πολλαπλών αντικειμένικών συναρτήσεων για τη βελτιστοποίηση του επιλεγμένου χαρτοφυλακίου. Τέλος, το τέταρτο υποσύστημα υποστηρίζει τη διαδικασία συγκριτικής αξιολόγησης των παραχθέντων χαρτοφυλακίων.

Επιπλέον, το πρώτο υποσύστημα αναπτύσσεται ως διαδικτυακή εφαρμογή. Η συγκεκριμένη πλατφόρμα προσφέρει ένα φιλικό γραφικό περιβάλλον χρήστη *GUI* και προσφέρει αποδοτικές υλοποιήσεις συγκεκριμένων πολυκριτήριων μεθόδων, παρέχοντας εκτεταμένες λύσεις για ένα ευρύ φάσμα προβλημάτων πολλαπλών κριτηρίων.

Αρχιτεκτονική συστήματος

Το πληροφοριακό σύστημα αναπτύχθηκε στη γλώσσα προγραμματισμού Python 3, γεγονός που το καθιστά διαθέσιμο για λειτουργικά συστήματα Windows, Linux και macOS. Επιπλέον, για την ανάπτυξη των επιμέρους λειτουργιών του συστήματος χρησιμοποιήθηκε ένας αριθμός βιβλιοθηκών της γλώσσας Python. Οι κυριότερες είναι οι: Pandas, NumPy, MatPlotlib και MIP. Στη συνέχεια δίνεται μια σύντομη επεξήγηση των παραπάνω βιβλιοθηκών:

Η βιβλιοθήκη *MatPlotlib* είναι η σημαντικότερη βιβλιοθήκη σχεδίασης και παρουσίασης γραφικών παραστάσεων της γλώσσας Python. Παρέχει γραφικές παραστάσεις υψηλής ποιότητας σε μια ποικιλία απο διαφορετικά φορμάτ ανάλογα με τις εκάστοτε ανάγκες. Η βιβλιοθήκη *Pandas* είναι μια βιβλιοθήκη ανοικτού κώδικα που παρέχει χρήσιμες δομές δεδομένων και εργαλεία ανάλυσης δεδομένων υψηλής απόδοσης για τη γλώσσα προγραμματισμού Python. Η βιβλιοθήκη *NumPy* είναι η θεμελιώδης βιβλιοθήκη για επιστημονική πληροφορική με την Python, καθώς περιέχει διάφορα χρήσιμα εργαλεία με κυριότερο τον πίνακα N-διαστάσεων ο οποίος παρέχει στο χρήστη μια σειρά από χρήσιμες λειτουργικότητες. Τέλος, η βιβλιοθήκη *MIP* είναι μια βιβλιοθήκη της Python για τη μοντελοποίηση και επίλυση προβλημάτων γραμμικού και μεικτού-ακέραιου προγραμματισμού.

Παρουσίαση πολυκριτήριας πλατφόρμας

Στο πλαίσιο του έργου της διπλωματικής, το πρώτο επίπεδο του πληροφοριακού συστήματος αναπτύχθηκε ως διαδικτυακή εφαρμογή. Αυτή η εφαρμογή περιλαμβάνει μια αποδοτική υλοποίηση μιας σειράς μεθόδων πολυκριτηριακής ανάλυσης αποφάσεων, όπως οι ELECTRE, PROMETHEE, MAUT και TOPSIS. Το έργο υλοποιήθηκε σε ένα από τα γνωστότερα Python frameworks με την ονομασία *Django*. Στην παράγραφο αυτή παρουσιάζονται μερικά από τα πιο σημαντικά χαρακτηριστικά της πλατφόρμας:

Η αρχική σελίδα του συστήματος είναι σχεδιασμένη ώστε να προσφέρει μια παρουσίαση του περιεχομένου της εφαρμογής όπως φαίνεται στην εικόνα 2.

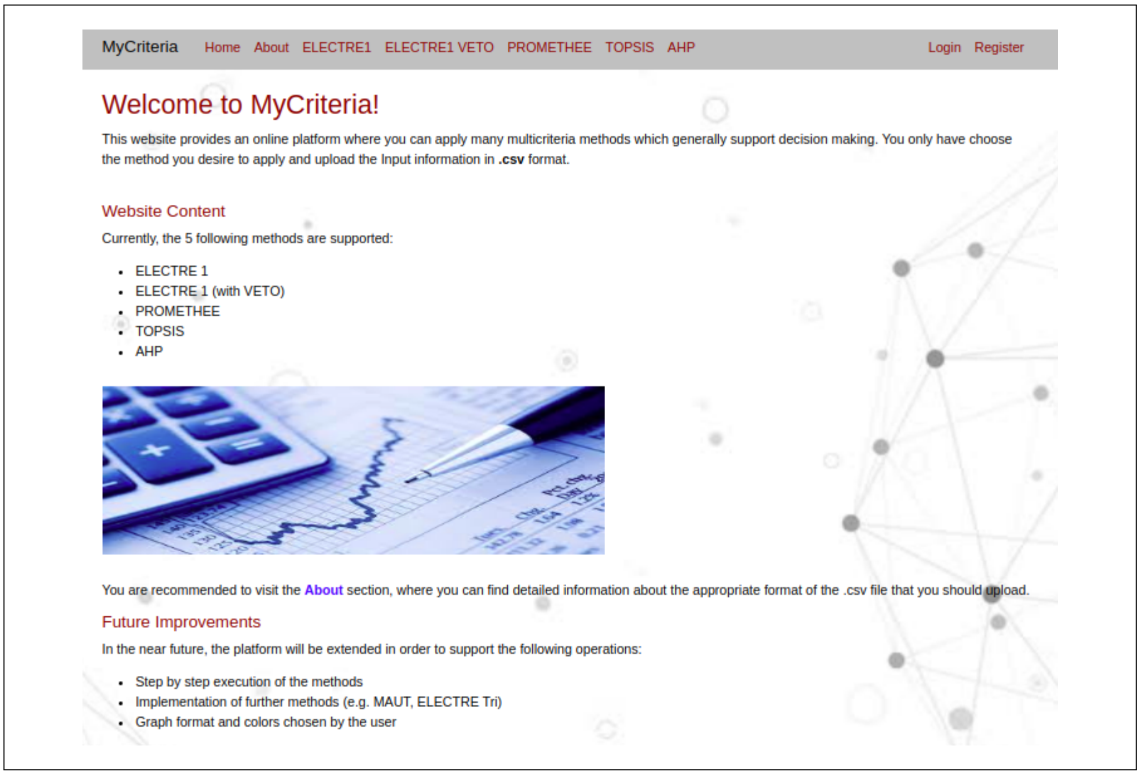


Figure 2: Αρχική Οθόνη

Ο χρήστης έχει τη δυνατότητα να επιλέξει οποιαδήποτε από τις υποστηριζόμενες πολυκριτήριες μεθόδους, αποκτώντας πρόσβαση στις λεπτομερείς πληροφορίες και τις απαιτήσεις εισόδου της επιλεγμένης μεθόδου. Στην εικόνα 3 παρουσιάζεται ενδεικτικά η σελίδα της μεθόδου PROMETHEE.

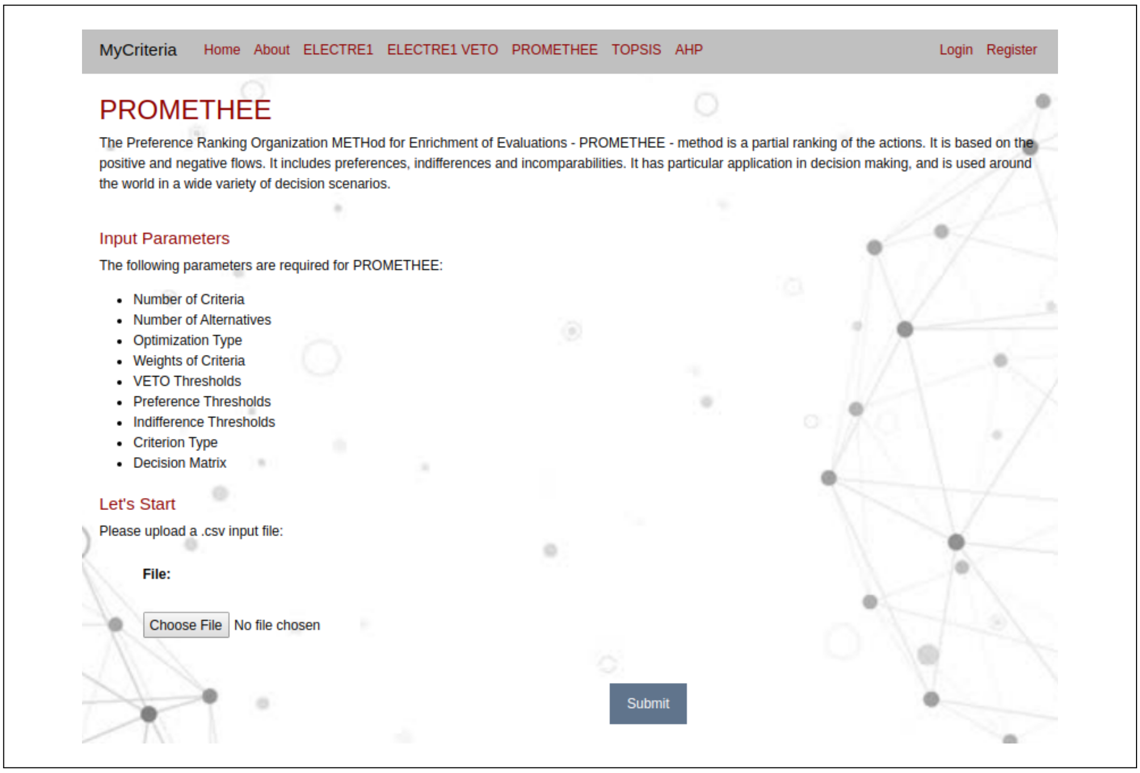
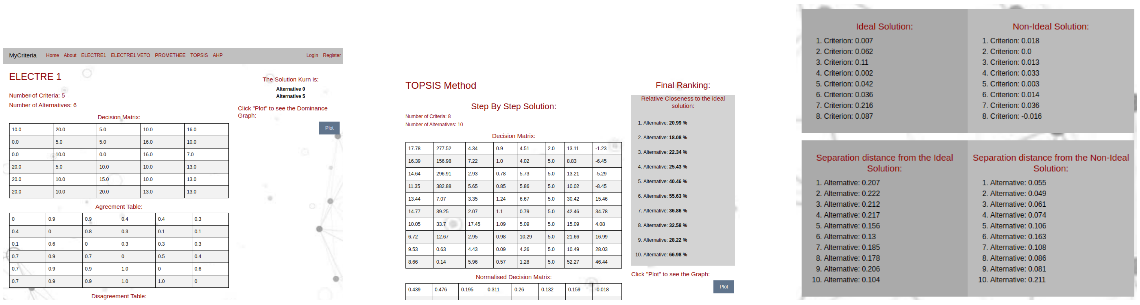


Figure 3: Οθόνη πολυκριτήριας μεθόδου

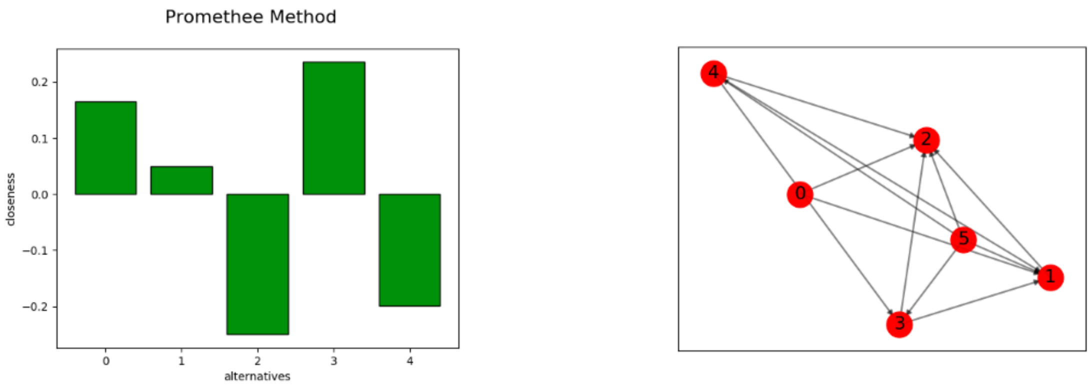
Τα δεδομένα εισόδου μπορούν να εισαχθούν στην πλατφόρμα σε αρχείο μορφής *.csv* ή *.xls*. Εκτός από τις γενικές πληροφορίες (εναλλακτικές λύσεις, κριτήρια, μήτρα αποφάσεων και βάρη), η κάθε μέθοδος περιλαμβάνει διαφορετικά στοιχεία εισόδου. Για παράδειγμα, στη μέθοδο PROMETHEE είναι απαραίτητο να καθοριστεί ο τύπος του κριτηρίου και τα κατάλληλα κατώφλια. Μετά τη διαδικασία εισαγωγής του αρχείου εισόδου στην πλατφόρμα, το επόμενο στάδιο περιλαμβάνει την παρουσίαση των αποτελεσμάτων. Στα σχήματα της εικόνας 4, παρουσιάζεται η οθόνη εξόδου μετά την εφαρμογή ορισμένων από τις υποστηριζόμενες μεθόδους.



(a) ELECTRE I (b) TOPSIS (i) (c) TOPSIS (ii)

Figure 4: Οθόνη παρουσίασης των αποτελεσμάτων

Τέλος, η απεικόνιση των αποτελεσμάτων μπορεί να πραγματοποιηθεί κλικάροντας την αντίστοιχη επιλογή στη σελίδα αποτελεσμάτων. Ο τρόπος απεικόνισης εξαρτάται από τη μέθοδο και προσαρμόζεται άμεσα στην κατηγορία της μεθόδου. Για παράδειγμα, για τις μεθόδους κατάταξης παράγεται ένα barplot με την τελική κατάταξη των εναλλακτικών επιλογών, ενώ για τις μεθόδους επιλογής παράγεται το γράφημα κυριαρχίας, παρουσιάζοντας ποιές εναλλακτικές κυριαρχούν έναντι άλλων, όπως παρουσιάζεται στην εικόνα 5.



(a) Μέθοδος κατάταξης

(b) Μέθοδος επιλογής

Figure 5: Οθόνη οπτικοποίησης των αποτελεσμάτων

Κεφάλαιο 6: Εφαρμογή προτεινόμενης μεθοδολογίας

Στα προηγούμενα κεφάλαια έγινε μια περιγραφή του προτεινόμενου μεθοδολογικού πλαισίου και επιπλέον παρουσιάστηκε το πληροφοριακό σύστημα που υλοποιεί την προτεινόμενη μεθοδολογία. Ωστόσο, είναι απαραίτητο η εγκυρότητα του μεθοδολογικού πλαισίου να δοκιμαστεί με πραγματικά δεδομένα. Συνεπώς, διεξήχθη μια εκτενής πειραματική εφαρμογή του προτεινόμενου μεθοδολογικού πλαισίου σε τέσσερις μεγάλες αγορές, συμπεριλαμβανομένων όλων των φάσεων από την επιλογή των χρεογράφων μέχρι και τη βελτιστοποίηση του χαρτοφυλακίου.

Χαρακτηριστικά πεδίου εφαρμογής

Η προτεινόμενη μεθοδολογία δοκιμάστηκε σε ένα σύνολο περίπου δύο χιλιάδων μετοχικών τίτλων. Ο χρονικός ορίζοντας της ανάλυσης ορίστηκε στα τρια ημερολογιακά έτη (2016 - 2018). Οι τίτλοι διαχωρίστηκαν ανάλογα με τον βιομηχανικό τομέα και το χρηματιστήριο στο οποίο ανήκουν. Οι οικονομικοί δείκτες για τους τίτλους λήφθηκαν από τη βάση δεδομένων *investing.com*. Ωστόσο, για ένα σημαντικό αριθμό χρεογράφων δεν υπήρχαν επαρκή στοιχεία. Συνεπώς, οι εταιρείες που δεν πληρούσαν τις απαιτήσεις (ελλείποντα δεδομένα, μηδενικές τιμές κλπ.) εξαιρέθηκαν από την πειραματική εφαρμογή. Στον πίνακα 1 καταγράφεται ο αριθμός μετοχικών τίτλων κάθε χρηματιστηρίου, χωρισμένοι ανάλογα με τον βιομηχανικό τους κλάδο.

Χρηματιστήριο	Βιομηχανικός κλάδος	Αριθμός τίτλων πειράματος	Αριθμός τίτλων με ανεπαρκή στοιχεία	Συνολικός αριθμός χρεογράφων
NYSE	technology	69	177	246
	energy	89	131	220
	financial	358	461	819
NASDAQ	technology	326	213	539
	energy	6	40	46
	financial	93	471	564
CAC 40	technology	50	91	141
	energy	7	8	15
	financial	33	24	57
Nikkei 225	technology	485	263	748
	energy	30	4	34
	financial	143	51	194

Table 1: Πληροφορίες δεδομένων εισόδου εφαρμογής

Αποτελέσματα

Προκειμένου να θεωρηθεί η προτεινόμενη μεθοδολογία ολοκληρωμένη και συνεπής, είναι απαραίτητο οι αποδόσεις των παραχθέντων χαρτοφυλακίων να συγκριθούν με τις αποδόσεις που προσφέρει η αγορά, σε μεταγενέστερο χρονικό ορίζοντα από τη χρονική στιγμή της επένδυσης. Μέσω αυτής της διαδικασίας πραγματοποιείται η επικύρωση των παραχθέντων αποτελεσμάτων.

Η διαδικασία της πιστοποίησης των αποτελεσμάτων πραγματοποιείται ως εξής: Αποδεικνύεται ότι τα χαρτοφυλάκια που προτείνονται στον επενδυτή απο την προτεινόμενη μεθοδολογία, παρουσιάζουν όμοια ή καλύτερη απόδοση, σε σχέση με τους δείκτες αναφοράς της αγοράς. Επομένως, η διαδικασία του ελέγχου των αποτελεσμάτων είναι βασισμένη σε δεδομένα τα οποία είναι μεταγενέστερα από το αρχικό δείγμα. Τα αποτελέσματα της διαδικασίας επικύρωσης των αποτελεσμάτων για τις τέσσερις αγορές παρουσιάζονται στις γραφικές παραστάσεις 6 - 9.

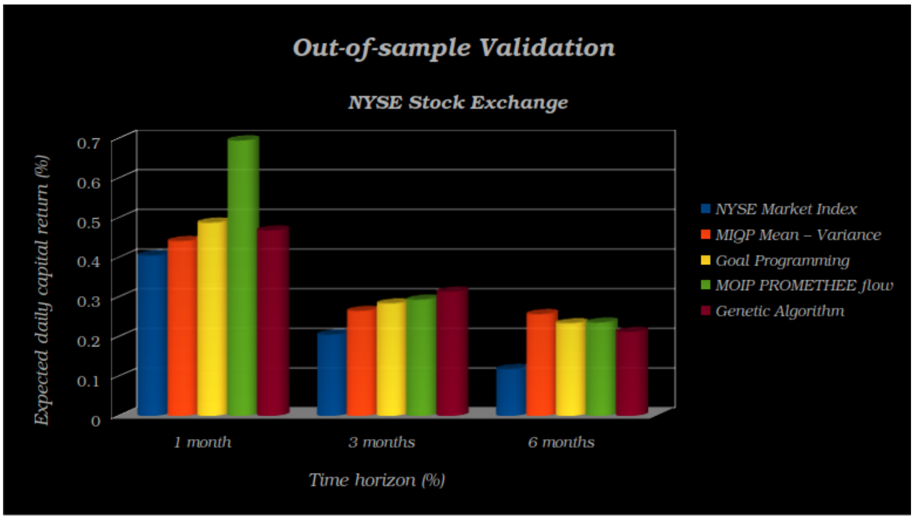


Figure 6: Συγκριτική γραφική παρουσίαση της μέσης ημερήσιας απόδοσης (%) για τα επιλεχθέντα χαρτοφυλάκια και το δείκτη αναφοράς της αγοράς (NYSE)

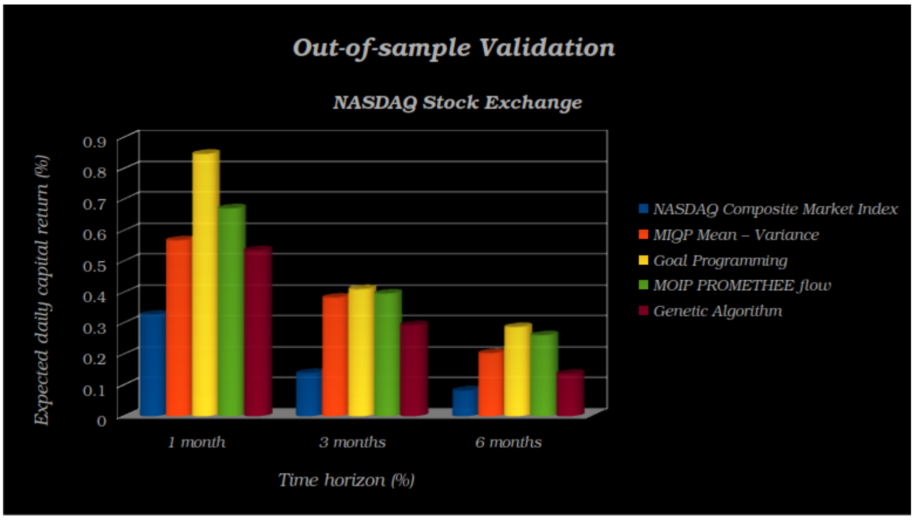


Figure 7: Συγκριτική γραφική παρουσίαση της μέσης ημερήσιας απόδοσης (%) για τα επιλεχθέντα χαρτοφυλάκια και το δείκτη αναφοράς της αγοράς (NASDAQ)

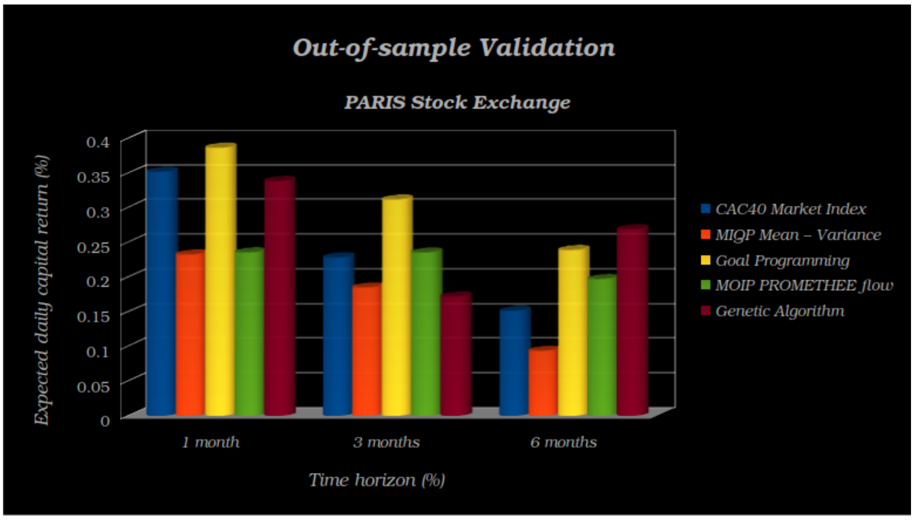


Figure 8: Συγκριτική γραφική παρουσίαση της μέσης ημερήσιας απόδοσης (%) για τα επιλεχθέντα χαρτοφυλάκια και το δείκτη αναφοράς της αγοράς (PARIS)

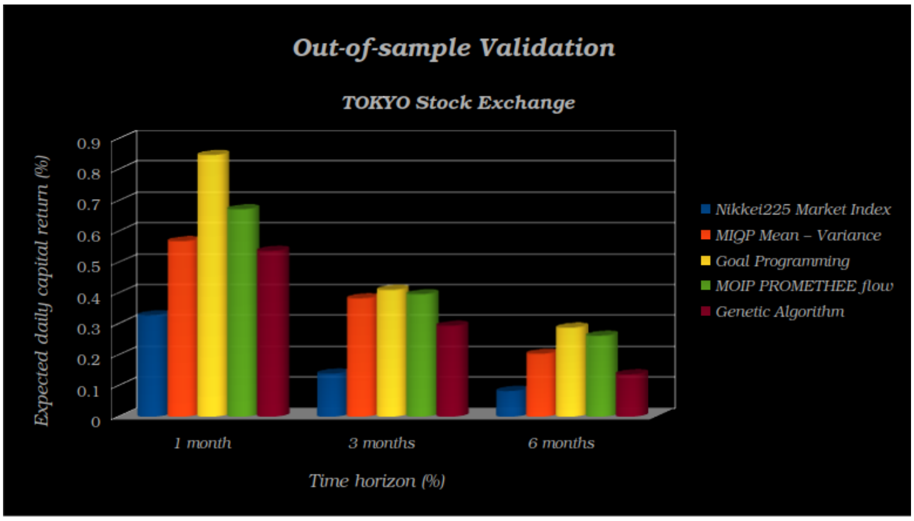


Figure 9: Συγκριτική γραφική παρουσίαση της μέσης ημερήσιας απόδοσης (%) για τα επιλεχθέντα χαρτοφυλάκια και το δείκτη αναφοράς της αγοράς (TOKYO)

Τα αποτελέσματα της διαδικασίας επαλήθευσης (out-of-sample validation process) ανέδειξε την αξιοπιστία της προτεινόμενης μεθοδολογίας, οδηγώντας σε ικανοποιητικά αποτελέσματα. Οι αποδόσεις των παραχθέντων χαρτοφυλακίων μέσω της προτεινόμενης μεθοδολογίας θεωρούνται ιδιαίτερα ανταγωνιστικές σε σύγκριση με τις αποδόσεις που προσέφερε η αγορά κατά το ίδιο χρονικό διάστημα. Μάλιστα, η αξία των αποτελεσμάτων της επαλήθευσης μεγεθύνεται δεδομένου ότι μια σειρά χρεογράφων εξαιρέθηκαν απο τη διαδικασία λόγω της έλλειψης επαρκών δεδομένων.

Κεφάλαιο 7: Συμπεράσματα και προοπτικές

Σήμερα, η ανάγκη για ολοκληρωμένα μεθοδολογικά πλαίσια και συστήματα υποστήριξης αποφάσεων είναι ισχυρότερη από ποτέ. Αυτές οι μεθοδολογίες πρέπει να συμπεριλαμβάνουν όλα τα αντικρουόμενα κριτήρια και τις αλληλεπιδράσεις μεταξύ τους, καθώς και την αβεβαιότητα της χρηματοπιστωτικής αγοράς και τα διαφορετικά προφίλ των φορέων λήψης αποφάσεων. Συνεπώς, αυτή η εργασία συμβάλλει στην αναγνώριση όλων των παραμέτρων του προβλήματος της διαχείρισης χαρτοφυλακίου και των αλληλεπιδράσεων μεταξύ όλων αυτών των παραγόντων. Σε αυτό το κεφάλαιο, παρουσιάζονται τα κύρια συμπεράσματα των προηγούμενων κεφαλαίων.

Συμπεράσματα

Το βασικό μοντέλο μέσου - διακύμανσης βασίζεται σε δύο κριτήρια (απόδοση και κίνδυνος), γεγονός που το καθιστά ανεπαρκές για την αντιμετώπιση ενός ρεαλιστικού προβλήματος. Αντιθέτως, μια ολοκληρωμένη προσέγγιση απαιτεί την ενσωμάτωση όλων των αντικρουόμενων κριτηρίων που επηρεάζουν την αγορά ασφάλειας. Κατά συνέπεια, το πρόβλημα διαχείρισης χαρτοφυλακίου είναι ένα πρόβλημα πολλαπλών κριτηρίων το οποίο απαιτεί ένα ολοκληρωμένο μεθοδολογικό πλαίσιο εδρασμένο σε πολλαπλά κριτήρια.

Έχοντας αναγνωρίσει όλες τις δυσκολίες του προβλήματος και τις ελλείψεις στις υπάρχουσες μεθοδολογίες, η τρέχουσα εργασία περιλαμβάνει την ανάπτυξη ενός ολοκληρωμένου μεθοδολογικού πλαισίου για τη διαχείριση χαρτοφυλακίου. Το πλαίσιο αυτό αποτελείται από δύο κύριες φάσεις: (i) τη φάση της επιλογής χαρτοφυλακίου και (ii) τη φάση της βελτιστοποίησης χαρτοφυλακίου.

Η μεθοδολογία διαχείρισης χαρτοφυλακίου τονίζει την ανάγκη για σύγχρονα πληροφοριακά σύστηματα οποία υλοποιούν αποτελεσματικά τα θεωρητικά μοντέλα. Ως εκ τούτου, στο πλαίσιο αυτού του έργου σχεδιάστηκε ένα σύστημα πληροφοριών το οποίο υλοποιεί τις προτεινόμενες πολυκριτήριες μεθόδους. Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε για την υλοποίηση είναι η Python 3, συνοδευόμενη από μερικές επιστημονικές βιβλιοθήκες όπως οι matplotlib, numpy και pandas. Επιπλέον, το υποσύστημα επιλογής χαρτοφυλακίου αναπτύχθηκε ως εφαρμογή web.

Τέλος, η προτεινόμενη μεθοδολογία εφαρμόστηκε επιτυχώς σε τέσσερις διεθνείς αγορές, σε τρείς από τους μεγαλύτερους βιομηχανικούς κλάδους. Η υπολογιστική προσπάθεια ήταν σημαντικά χαμηλότερη σε σύγκριση με τις συμβατικές μεθόδους, ενώ τα αποτελέσματα της προτεινόμενης μεθοδολογίας κατα τη διαδικασία της της πιστοποίησης ήταν ιδιαιτέρως ενθαρρυντικά.

Προοπτικές

Η παρούσα εργασία δημιουργεί ενδιαφέρουσες μελλοντικές προοπτικές έρευνας, οι κυριότερες εκ των οποίων παρουσιάζονται παρακάτω:

Υλοποίηση πλήρους πληροφοριακού συστήματος ως διαδικτυακή εφαρμογή: Στο πλαίσιο της παρούσας διπλωματικής, σχεδιάστηκε ένα ολοκληρωμένο πληροφοριακό σύστημα υποστήριξης αποφάσεων. Μάλιστα, το υποσύστημα επιλογής χαρτοφυλακίου χρεογράφων με πολυκριτήριες μεθόδους αναπτύχθηκε ως διαδικτυακή εφαρμογή. Μια σημαντική προοπτική σε αυτό τον τομέα είναι η ανάπτυξη ολόκληρου του συστήματος ως διαδικτυακή εφαρμογή. Η προέκταση αυτή δεν είναι ιδιαιτέρως δύσκολο να υλοποιηθεί, καθώς η ανάπτυξη των υπολοίπων υποσυστημάτων αποτελεί γενίκευση της υπάρχουσας διαδικτυακής εφαρμογής.

Διασύνδεση του πληροφοριακού συστήματος με εμπορικές εφαρμογές: Το αναπτυχθέν πληροφοριακό σύστημα υλοποιεί την προτεινόμενη μεθοδολογία διαχείρισης μετοχικών χαρτοφυλακίων, συνιστώντας ένα ολοκληρωμένο εργαλείο υποστήριξης επενδυτικών αποφάσεων. Σε αυτό το πεδίο η διαφαινόμενη μελλοντική προοπτική αφορά στη διασύνδεση του πληροφοριακού συστήματος με υπάρχουσες εμπορικές εφαρμογές. Μέσω της διασύνδεσης αυτής θα καταστεί πληρέστερη η διαδικασία υποστήριξης της επενδυτικής απόφασης για κάθε πελάτη. Βεβαίως η προοπτική αυτή χρειάζεται ιδαίτερη προσοχή, καθώς γεννόνται θέματα ασφάλειας της εφαρμογής.

Chapter 1

Introduction

1.1 Introduction

Nowadays, one of the major problems of the financial sector is the creation and management of an efficient investment portfolio under the complex environment of globalised society, rapidly increasing competition and sweeping economic changes at national and international level. Generally, investment portfolio is a group (portfolio) of assets, which were acquired based on a specific economic objective, aiming to generate profit for the investor.

Until the 1950s, the concept of portfolios was completely different. Investing in equities was a gambling process as there was insufficient financial data available and few people had realized the importance of investment management. Investors usually focused on the opportunities offered by each equity and not in a return-risk relationship.

The above situation radically changed since 1952, when nobel laureate H. Markowitz published his research work under the title "Portfolio Selection" [11], where he introduced the mathematical relation between the return and the risk of a security. According to Markowitz mean-variance model, a combination of different kinds of equities is less risky than owning only one type. Subsequently, investors started creating portfolios that favored specific investment styles and preferences, using the mean-variance model or other models which tried to expand it and extinguish its weaknesses. Therefore, nowadays the process of creating and managing equity portfolios has significantly developed and cultivated.

However, it is well known that the global economy has historically been shaken by strong fluctuations, making equities one of the most vulnerable markets. Equity portfolios are the most risky market placement for two main reasons, according to Xidonas (2010) [27]. Firstly, there is no possibility of differentiating part of the risk, investing in fixed-income securities and deposit or derivative products. Secondly, the process of equity portfolio management is extremely difficult due to the existence of a large number of equities traded on Stock markets. This fact renders necessary the investigation of thousands of securities, which are available as investment choices.

Equity portfolio management is a very complex problem, as it focuses on three different levels of decision-making:(i) selecting equity securities which encapsulate the best investment prospects, (ii) distributing the available capital in order to achieve optimal portfolio composition and (iii) comparative evaluation of the constructed portfolios

Besides, the problem of equity portfolio management is linked to three other fundamental parameters that affect each decision-making process: (i) uncertainty, (ii) the existence of multiple criteria and (iii) the profile and preferences of the decision-maker (DM)

Last but not least, another crucial parameter is the existence of many stakeholders because of the complexity of modern economies and markets. In fact, the whole process is made more difficult by the fact that these stakeholders usually have different, or even conflicting, interests. More specifically, the entities that constitute the environment of this particular problem, can be grouped into four categories: (a) entities which are associated with the supervision of the market, (b) companies listed on the stock market, (c) institutional and private investors and (d) investment service providers.

In conclusion, the above parameters demonstrate the enormous complexity and uncertainty in the financial decision-making process and imply the need for appropriate indicators and supportive decision tools. These tools are intended to replace decision-making based exclusively on empirical approaches with modern methods of analysis, resulting in a more efficient treatment of investment risks and equity portfolios management.

1.2 Thesis Target and Objective

As discussed in the above section, the need for development of integrated methodological frameworks and decision support systems today is stronger than ever. These frameworks should encapsulate all the criteria and the interactions between them, as well as the uncertainty of the financial market and the different profiles and needs of the stakeholders.

The origin of this research effort is the classic mean-variance theory of Markowitz. This approach is very useful but not sufficient in order to effectively address the problem of equity portfolios management. Therefore, the proposed framework aspires to extinguish the weaknesses of the mean-variance model and to overcome the existing computational difficulties

The subject of this thesis is the development of an integrated decision-support methodology for equity portfolios management, in the context of strong volatility and growing uncertainty in the modern financial environment.

The purpose of this thesis is the identification of all the parameters of the problem, the extensive analysis of the interactions between them and finally the configuration of a transparent and consistent decision-support framework.

Additionally, the thesis includes the development of an integrated portfolio management information system, which implements the proposed methodology. The validity of the information system was succesfully verified through an extensive experimental application on four major international stock exchanges.

1.3 Thesis Contribution and Value

Generally, the thesis contributes to the scientific society an integrated methodological framework for equity portfolio management, as well as a modern decision-support information system. Additionally, each of the individual steps of the methodology could be successfully applied, even separately from the entire framework.

More specifically, the contribution of the thesis is described in the following paragraph 1.1:

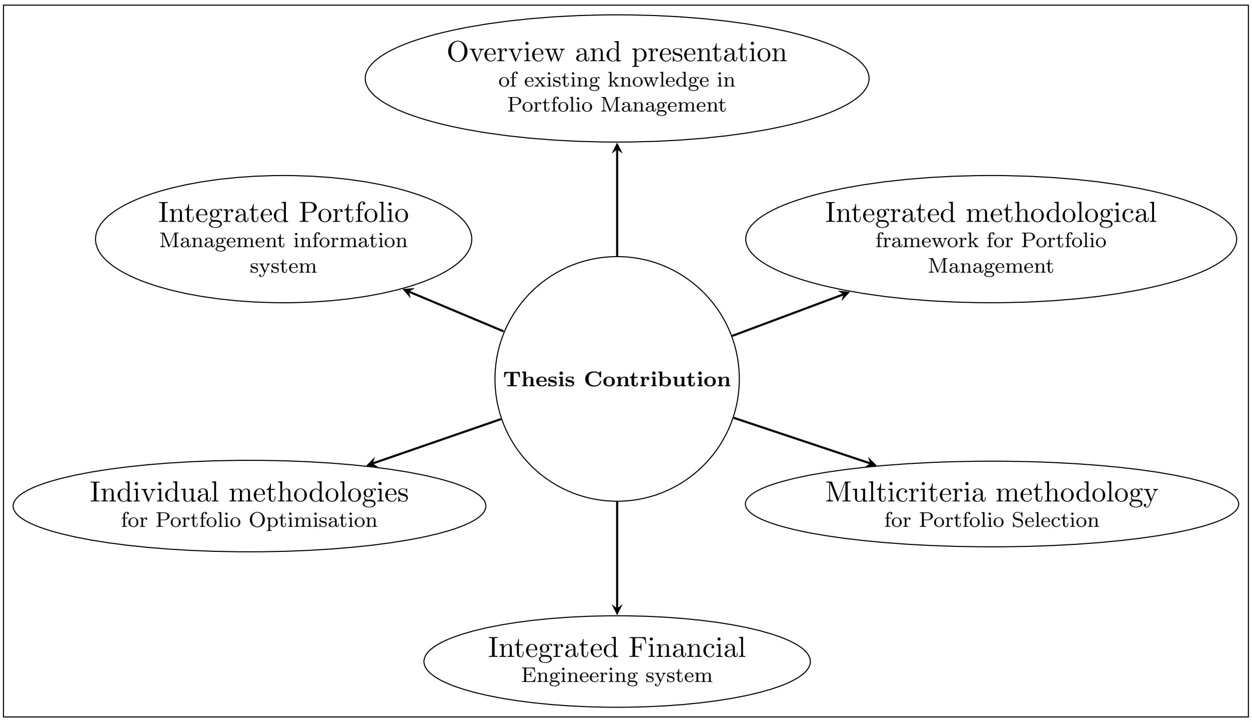


Figure 1.1: Thesis Contribution

The thesis contributes an elaborate overview of the existing knowledge in the field of Portfolio Management, including a wide range of tecnhical terminology and presenting the basic concepts of modern portfolio theory (MPT). Additionally, a detailed discussion of multicriteria decision analysis (MCDA) is attempted.

An integrated methodological framework is developed which aspires to include the whole procedure of portfolio management. The methodological framework consists of two basic subsystems: (a) the portfolio selection subsystem, (b) the portfolio optimisation subsystem.

The first phase of the process is based on a multicriteria decision-making methodology for security selection. This process is based on four fundamental MCDA ranking methods which are combined in order to support the selection of the best securities. In this thesis project, these MCDA methods are explained in detail and additionally pseudocode for these methods is included.

As part of the proposed methodology, an integrated subsystem for security returns and statistical indexes is developped. This subsystem includes some useful visualisation tools for inspecting security values and returns as well as a variety of statistical indices such as correlation, kurtosis etc.

The second phase of the methodology demands continuous mathematical optimisation methods. In this project, we develop various models for portfolio optimisation, including a mean-variance model, a genetic algorithm approach, a goal-programming mathematical optimisation method and a multidimensional MCDA optimisation technique involving the PROMETHEE flow.

Finally, as part of this project, a complete information system is developed in order to support the whole procedure. Additionally, the MCDA subsystem was designed as a web application which implement a variety of multicriteria methods with step-by-step detailed solutions. This application provides a useful, user-friendly tool for decision support with many alternatives.

1.4 Thesis Structure

The thesis consists of six chapters and two appendices. Here is a brief description of their contents.

Chapter 1
In the 1st chapter of the thesis, a brief introduction to the problem was made, defining the main attributes and the historical background. The thesis target and objective were defined, as well as its contribution to the scientific society. Finally, there was a description of the thesis structure.

Chapter 2
In the 2nd chapter of the thesis, we make an introduction to the portfolio management problem. The chapter begins with some definitions of the fundamental formulas and the description of the concept of diversification. Additionally, there is a presentation of the portfolio optimisation problem with and without shrt sales. Finally, the case of the risk free asset is introduced. The chapter also contains some basic proofs of the fundamental equations for completeness reasons.

Chapter 3
In the 3rd chapter of the thesis, there is a presentation of the related methodologies of the thesis. The chapter is divided in two parts. In the first part, there is an introduction to the discrete multiple-criteria decision analysis methods, including some basic definitions and a historical overview. In the second part, there is an introduction to multiobjective mathematical programming. There is an introduction to linear, quadratic and integer programming. The second part of the chapter discusses the concepts of multiobjective programming, goal programming and genetic algorithms.

Chapter 4
In the 4th chapter of the thesis, we present the proposed methodology. Initially, there is an overview of the methodology, including some illustrative diagrams. The two phases of the methodology are described in detail. The first phase discusses the problem of MCDA portfolio selection, including pseudocode of the ranking methods. The second phase presents the methodologies for multiobjective portfolio optimisation.

Chapter 5
In the 5th chapter of the thesis, we present the information system that was developped as part of the thesis. The chapter covers all the tools and libraries that were used. Additionally, it includes the basic UML diagrams which describe the information system. In the second part of the chapter, there is a brief presentation of the MCDA web application including pictures from the front-end and explanation of

the user interaction surface. Finally, in the last part a small part of the source code is presented with jupiter notebook, which demonstrates the security visualisation and financial statistics subsystem.

Chapter 6
In the 6th chapter of the thesis, we demonstrate a small part of the empirical testing results. More specifically, we explicitly present the results of each step of the methodological framework. The main volume of the empirical testing is placed in the appendix.

Chapter 7
Finally, in the 7th chapter of the thesis, a conclusion of the whole project is made and the future prospects are discussed.

Appendix A
In appendix A, we present the main volume of the source code in a jupiter notebook, accompanied by step-by-step results, in order to explain the code in a more understable way. The source code of the four MCDA methods, as well as the optimisation techniques is placed in the appendix, accompanied by short comments.

Appendix B
In appendix B, the extensive results of the empirical testing are presented. The results presented in this part refer to three of the largest industrial sectors (technological, energy and financial) and four major stock exchanges (NYSE, NASDAQ, Paris, Tokyo).

CHAPTER 2

The Portfolio Management Problem

2.1 Introduction

In this chapter there is a discussion of the portfolio management problem and more specifically the mean-variance methodology developped by Harry Markowitz (1952, 1959) [11], [12]. The problem of portfolio composition was introduced as a quadratic mathematical programming problem. Since then many scientists have attempted to improve this methodology and cure its weaknesses using a variety of optimisation techniques and other operational research methods. The presentation of the mean-variance methodological framework is developed in four sections.

In section 1 there is a brief discussion of the basic concepts that constitute the problem. The most significant terms are defined, such as the return and the risk both in case of a single security, as well as in the general case of a portfolio of securities. Additionally, the cases of a portfolio including two and three securities are presented, offering an introduction to the problem.

In section 2 the fundamental principle of diversification is presented. A presentation of the two different components of risk (systematic and non-systematic risk) is attempted, analysing the factors that make necessary the endorsement of a diversified strategy. Finally, some analytical proofs for portfolio risk are given.

In section 3 there is a detailed description of the problem of portfolio optimisation. The concepts of efficient portfolios and efficient frontier are introduced. Both the case that short sales are allowed and the case that short sales are restricted, are discussed. Additionally, a theoritical proof for the detection of the efficient frontier is given in case of short sales.

In section 4 the concept of the risk free security is introduced. The analysis is again divided according to the existance of short sales. Finally, two different proofs are developed in case of short sales, offering two different views of the same problem.

2.2 Fundamental Formulas

A legal contract which allows the investor to receive some future economic benefits under specific and clearly formulated conditions is called *security*. *Common stocks* or *equities* are a subcategory of securities which provide the investor the right to participate in the profits of the company.

The percentage variation of an investment's value over a given period of time is defined as *security arithmetic return*. Let S_t be the value of a security at time t. The arithmetic return of a security is defined as follows:

$$r_t = \frac{S_t - S_{t-1}}{S_{t-1}} \tag{2.1}$$

Given the arithmetic return for a specific number of periods T, the *expected return* $E(r)$, also known as mean return is given by:

$$E(r) = \frac{1}{T}\sum_{t=1}^{T} r_t \tag{2.2}$$

Any deviation from the expected return is considered as *risk*. The typical measure of risk that is used in security markets is the standard deviation of a security's return over a number of periods. Therefore, risk σ can be measured with the variance criterion, as follows:

$$\sigma^2 = \frac{1}{T}\sum_{t=1}^{T}[r_t - E(r)]^2 \tag{2.3}$$

The term *portfolio* refers to any combination of financial assets such as stocks, bonds and cash. Each asset participates in the portfolio in some proportion which is determined by the value of the asset relatively to the total value of the portfolio. In the following section, the concept of security portfolios is discussed.

Let P be a portfolio consisting of m securities. The portfolio return $E(r_P)$ and risk σ_P^2 are defined as follows:

$$E(r_P) = \sum_{i=1}^{m} w_i E(r_i) \tag{2.4}$$

$$\sigma_P^2 = \sum_{i=1}^{m} w_i^2 \sigma_i^2 + \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} w_i w_j \sigma_{ij} \tag{2.5}$$

where $E(r_i)$, σ_i^2 are the expected return and the variance of the i_{th} security, σ_{ij} is the covariance between securities i and j and w_i is the participation percentage of security i in the portfolio.

The covariance between the returns of securities i and j is defined as follows :

$$\sigma_{ij} \equiv COV(r_i, r_j) \doteq \frac{1}{T}\sum_{i=1}^{T}[r_{it} - E(r_i)][r_{jt} - E(r_j)] \tag{2.6}$$

The standard deviation of the portfolio is called *portfolio volatility*:

$$\sigma_p = \sqrt{\sigma_p^2} \tag{2.7}$$

For a portfolio which consists of two securities with expected returns r_1 and r_2, variances σ_1^2 and σ_2^2, respectively and covariance σ_{12} the above definitions are formed as follows:

Portfolio return:

$$\mathrm{E}(r_p) = w_1 \,\mathrm{E}(r_1) + w_2 \,\mathrm{E}(r_2) = w_1 \,\mathrm{E}(r_1) + (1 - w_1) \,\mathrm{E}(r_2). \tag{2.8}$$

Portfolio variance:

$$\sigma_p^2 = w_1^2\sigma_1^2 + w_2^2\sigma_2^2 + 2w_1w_2\sigma_{12} \tag{2.9}$$

For a portfolio which consists of three securities with expected returns r_1, r_2 and r_3, variances σ_1^2, σ_2^2 and σ_3^2 and covariances σ_{12}, σ_{13} and σ_{23} the above definitions are formed as follows:

Portfolio return:

$$\mathrm{E}(r_p) = w_1 \,\mathrm{E}(r_1) + w_2 \,\mathrm{E}(r_2) + w_3 \,\mathrm{E}(r_3) \tag{2.10}$$

Portfolio variance:

$$\sigma_p^2 = w_1^2\sigma_1^2 + w_2^2\sigma_2^2 + w_3^2\sigma_3^2 + 2w_1w_2\sigma_{12} + 2w_1w_3\sigma_{13} + 2w_2w_3\sigma_{23} \tag{2.11}$$

2.3 Diversification

According to Doumpos [25], a portfolio's risk can be reduced by holding combinations of securities which are not positively correlated. Thus, the portfolio variance is reduced. Holding a diversified portfolio of assets results in less significant exposure to individual asset risk. Therefore, *diversification* allows the investor to achieve the same portfolio expected return with a reduced percentage of risk. In the following paragraph, the significance of diversification strategies is explained.

On the one side, the portfolio expected return is a linear function of the individual securities which constitute the portfolio. As a consequence, the *maximum return portfolio* is the portfolio that consists of the security with the greatest return with proportion equal to 1. On the other side, the portfolio's risk is a non-linear function of the portfolio's securities proportions.

Assuming an equally distributed portfolio of infinite securities the following two observations indicate the benefits of diversification:

Uncorrelated securities

Firstly, assuming the fact that an investor could build a portfolio consisting of infinite uncorrelated securities, then the portfolio risk can be totally eliminated, according to the following proof:

Proof. Let σ_P^2 be the portfolio risk. For any pair of of securities i and j which are totally uncorrelated, the following applies: $\sigma_{ij} = 0$. In this case the portfolio risk is:

$$\sigma_P^2 = \sum_{i=1}^{m} w_i^2 \sigma_i^2 \tag{2.12}$$

Assuming an equally distributed portfolio consisting of m uncorrelated securities participating with weighting factor $w_i = 1/m$:

$$\lim_{m\to\infty} \sigma_P^2(m) = \lim_{m\to\infty} \sum_{i=1}^{m} \left(\frac{1}{m}\right)^2 \sigma_i^2 = \lim_{m\to\infty} \frac{1}{m}\overline{\sigma^2} = 0 \tag{2.13}$$

□

Correlated securities

Secondly, assuming the -most realistic- case that an investor could build a portfolio consisting of infinite securities, which are not uncorrelated, then the portfolio risk is defined by the securities correlations, because the risk of each individual security is eliminated.

Proof. Let σ_P^2 be the portfolio risk of an equally distributed portfolio consisting of m securities participating with weighting factor $w_i = 1/m$:

$$\begin{aligned}\lim_{m\to\infty}\sigma_P^2(m) &= \lim_{m\to\infty}\left[\sum_{i=1}^{m}\left(\frac{1}{m}\right)^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,j\neq i}^{m}\left(\frac{1}{m}\right)^2\sigma_{ij}\right] \\ \lim_{m\to\infty}\sigma_P^2(m) &= \lim_{m\to\infty}\left(\frac{1}{m}\overline{\sigma^2} + \frac{m-1}{m}\overline{\sigma_{ij}}\right) = \overline{\sigma_{ij}}\end{aligned} \tag{2.14}$$

□

As a conclusion, if the investor has the opportunity to compose a portfolio including infinite securities, the portfolio risk is determined by the correlations of the securities, while the individual risk of each security extinguishes. This fact signifies the necessity of a diversification strategy. Additionally, the total portfolio risk can be analysed in two components:

The first component is called *non-systematic risk* It reflects the risk of each individual security and it is not affected by the behaviour of the other securities. This component can be eradicated by an appropriate diversification strategy.

The second component is called *systematic risk*. If a portfolio's systematic risk is greater than 1, then it is expected to have higher volatility than the market. On the contrary, if a portfolio's systematic risk is lower than 1, then it is expected to have lower volatility than the market. If a portfolio has zero systematic risk then it is not affected by the market. The market is usually represented by a well known index like *S&P 500* or *RUSSELL 2000 INDEX*. This component of risk is not possible to be eradicated with diversification atrategy.

2.4 Portfolio Optimisation

The observations made in the above section lead to the conclusion that a diversification strategy is necessary for the construction of a solid portfolio. The first major methodological framework was developed by H. Markowitz in 1952 introducing the concept of the *efficient portfolio*. According to Markowitz definition, a portfolio P is efficient if and only if there is no other portfolio P' such that $E(r_{P'}) \geq E(r_P)$ and $\sigma_{P'} \leq \sigma_P$, given that at least one inequality is strict. The set including all the efficient portfolios is called *efficient frontier*.

As indicated in figure 2.1, a feasible portfolio is any portfolio with proportions summing to one. As stated by Benninga [20], the set of all feasible portfolios is called feasible set and it is depicted as the area inside and to the right of the curved line. All portfolios which have minimum variance for a given mean return are called envelope portfolios and they are depicted on the envelope of the feasible set. Finally all portfolios which have maximum return given the portfolio variance are called efficient portfolios and are depicted by the dense line in figure 2.1.

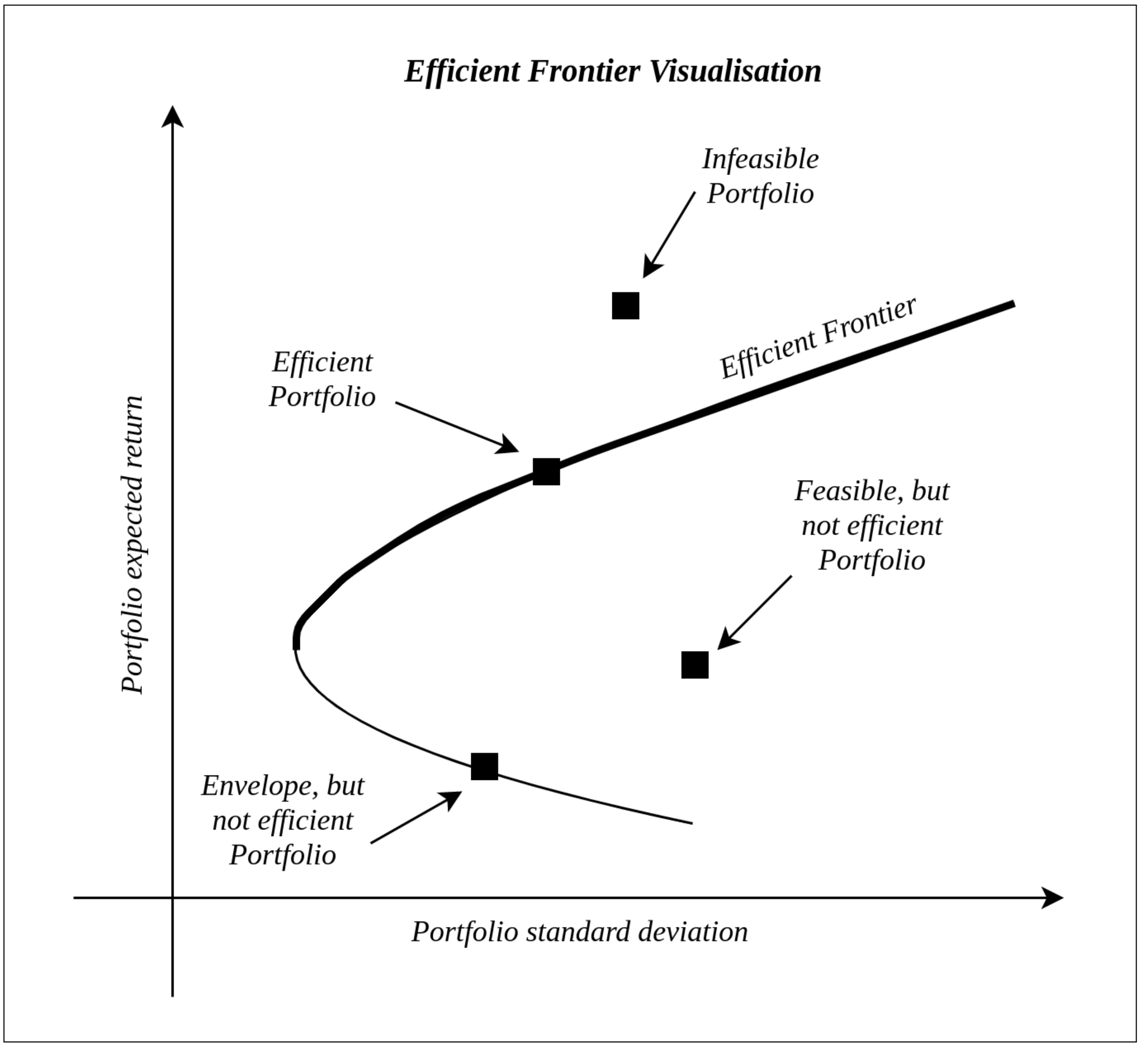


Figure 2.1: Visualisation of the efficient frontier

In this point we present the suitable techniques for portfolio optimisation considering two main cases: (a) when short sales are allowed and (b) when short sales are restricted:

Short sales allowed

In the first case, short sales are allowed, thus not constraining the weighting factors to be positive. The problem is defined as follows:

$$\begin{array}{ll} \min\limits_{\mathbf{w}} & \sigma_P^2 = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} \\ \max\limits_{\mathbf{w}} & E(r_P) = \mathbf{r}^T\mathbf{w} \\ \text{s.t.} & \mathbf{e}^T\mathbf{w} \;=\; 1 \\ & \quad\;\mathbf{w} \;\in\; \mathbb{R} \end{array} \tag{2.15}$$

The suggested approach is to transform the expected return R into a parameter and solve the problem of risk minimisation. This is a quadratic programming problem with linear contraints and can be solved using the Lagrange multipliers λ, leading to a linear system of equations.

$$\begin{bmatrix} 2V & -r \\ r^T & 0 \end{bmatrix} * \begin{bmatrix} w \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ R \end{bmatrix} \tag{2.16}$$

The following proof, given by Doumpos [25] describes the solution of the initial problem.

Proof. Let us consider a portfolio consisting of m securities. The purpose of this analysis is the composition of the optimal risk portfolio, given the expected return R. In this case the following problem is formulated:

$$\begin{array}{ll} \min\limits_{\mathbf{w}} & \sigma_P^2 = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} \\ \text{s.t.} & \mathbf{r}^T\mathbf{w} \;=\; R \\ & \mathbf{e}^T\mathbf{w} \;=\; 1 \\ & \quad\;\mathbf{w} \;\in\; \mathbb{R} \end{array} \tag{2.17}$$

where $\mathbf{e}$ represents the unary column-vector: $\mathbf{e} = (1, 1, \ldots, 1)^T$, $\mathbf{r}$ is the expected return vector and $\mathbf{V}$ is a positive definite matrix. This hypothesis restricts all securities to include some risk, resulting in a strictly convex risk function.

Let λ_1 and λ_2 be the Lagrange multipliers of the two constraints, the following function is formulated:

$$\mathscr{L} = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} + \lambda_1(1 - \mathbf{e}^T\mathbf{w}) + \lambda_2(R - \mathbf{r}^T\mathbf{w}) \tag{2.18}$$

In order to maximise this function its partial derivatives are set equal to zero:

$$\left\{\begin{array}{l} \dfrac{\partial \mathscr{L}}{\partial \mathbf{w}} = 0 \Rightarrow \;\; \mathbf{V}\mathbf{w} - \lambda_1 \mathbf{e} - \lambda_2 \mathbf{r} = 0 \\ \dfrac{\partial \mathscr{L}}{\partial \lambda_1} = 0 \Rightarrow \;\; \mathbf{e}^T \mathbf{w} = 1 \\ \dfrac{\partial \mathscr{L}}{\partial \lambda_2} = 0 \Rightarrow \;\; \mathbf{r}^T \mathbf{w} = R \end{array}\right\} \tag{2.19}$$

The first of the above equations gives:

$$\mathbf{w} = \lambda_1 \mathbf{V}^{-1}\mathbf{e} + \lambda_2 \mathbf{V}^{-1}\mathbf{r} \tag{2.20}$$

Substituting to the other two equations, the following system of linear functions is formulated:

$$\left\{\begin{array}{l} \lambda_1 \mathbf{e}^T \mathbf{V}^{-1}\mathbf{e} + \lambda_2 \mathbf{e}^T \mathbf{V}^{-1}\mathbf{r} = 1 \\ \lambda_1 \mathbf{r}^T \mathbf{V}^{-1}\mathbf{e} + \lambda_2 \mathbf{r}^T \mathbf{V}^{-1}\mathbf{r} = R \end{array}\right\} \tag{2.21}$$

Given that V^{-1} is a symmetric matrix, we take $\mathbf{e}^T\mathbf{V}^{-1}\mathbf{r} = \mathbf{r}^T\mathbf{V}^{-1}\mathbf{e}$. Therefore, let $a = \mathbf{e}^T\mathbf{V}^{-1}\mathbf{e}$, $b = \mathbf{e}^T\mathbf{V}^{-1}\mathbf{r}$ and $c = \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r}$, the linear system is transformed as follows:

$$\left\{\begin{array}{l} a\lambda_1 + b\lambda_2 = 1 \\ b\lambda_1 + c\lambda_2 = R \end{array}\right\} \tag{2.22}$$

The solution of the linear system is given by the following equations:

$$\begin{aligned} \lambda_1 &= \frac{c - bR}{ac - b^2} \\ \lambda_2 &= \frac{aR - b}{ac - b^2} \end{aligned} \tag{2.23}$$

The optimal portfolio composition is determined substituting λ_1 and λ_2 in (2.20). The portfolio risk is computed as follows:

$$\begin{aligned} \mathbf{V}\mathbf{w} - \lambda_1 \mathbf{e} - \lambda_2 \mathbf{r} = 0 & \quad \Rightarrow \\ \mathbf{w}^T\mathbf{V}\mathbf{w} - \lambda_1 \mathbf{w}^T\mathbf{e} - \lambda_2 \mathbf{w}^T\mathbf{r} = 0 & \quad \Rightarrow \\ \sigma_p^2 - \lambda_1 - \lambda_2 R = 0 & \quad \Rightarrow \\ \sigma_p^2 = \frac{aR^2 - 2bR + c}{ac - b^2} & \end{aligned} \tag{2.24}$$

The constants a, b, c are independent from the expected return R. In order to determine the *global minimum variance portfolio (GMVP)*, the constraint of the expected return R should be aborted. In this case the langrangian multipliers are $\lambda_2 = 0 \Rightarrow R = b/a$ and thus $\lambda_1 = 1/a$. The GMVP portfolio composition and the minimum portfolio risk are, respectively, calculated:

$$\mathbf{w} = \lambda_1 \mathbf{V}^{-1}\mathbf{e} + \lambda_2 \mathbf{V}^{-1}\mathbf{r} = \frac{1}{a}\mathbf{V}^{-1}\mathbf{e} = \frac{\mathbf{V}^{-1}\mathbf{e}}{\mathbf{e}^T\mathbf{V}^{-1}\mathbf{e}}$$

$$\sigma_p^2 = \lambda_1 + \lambda_2 R = \frac{1}{a}$$

□

Short sales restricted

The general approach assumes that there is a short sales restriction, thus allowing the portfolio proportions vary in range $[0, 1]$. The original portfolio optimisation problem is formulated as follows:

$$\begin{aligned}
\max_{w} \quad & E(r_P) = \sum_{i=1}^{m} w_i E(r_i) \\
\min_{w} \quad & \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j \neq i}^{m} w_i w_j \sigma_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} w_i = 1 \\
& w_i \geq 0, \qquad i = 1, 2, \ldots, m
\end{aligned} \tag{2.25}$$

In matrix form the problem is formulated as follows:

$$\begin{aligned}
\max_{\mathbf{w}} \quad & E(r_P) = \mathbf{r}^T \mathbf{w} \\
\min_{\mathbf{w}} \quad & \sigma_P^2 = \tfrac{1}{2} \mathbf{w}^T \mathbf{V} \mathbf{w} \\
\text{s.t.} \quad & \mathbf{e}^T \mathbf{w} = 1 \\
& \mathbf{w} \geq 0
\end{aligned} \tag{2.26}$$

where $\mathbf{V}$ is the variance-covariance matrix, $\mathbf{r}$ is the return column vector, $\mathbf{e}$ is the unary vector and $\mathbf{w}$ is the weighting factor vector.

This is a quadratic programming problem with linear restrictions. In 1956, Markowitz developed a general procedure for quadratic problems that can handle additional linear constraints and determine the entire set of efficient portfolios. This procedure was called *the critical line method*. An alternative approach for the specification of the efficient frontier is to solve the problem of risk minimisation, parametrically on the expected portfolio return, as follows:

$$\begin{aligned}
\min_{\mathbf{w}} \quad & \sigma_P^2 = \tfrac{1}{2} \mathbf{w}^T \mathbf{V} \mathbf{w} \\
\text{s.t.} \quad & \mathbf{r}^T \mathbf{w} = R \\
& \mathbf{e}^T \mathbf{w} = 1 \\
& \mathbf{w} \geq 0
\end{aligned} \tag{2.27}$$

where parameter R is the predefined expected portfolio return. Varying the expected return R between the edge values of *minimum variance portfolio* and *maximum return portfolio*, results in the determination of all points of the efficient frontier.

2.5 The Risk Free Asset

A security which has a certain rate of return, free of the various possible sources of risk is called *risk free security*. Given the existance of a risk free security , the investor would be interested to compose a portfolio FP which combines the risk free security F with a risk portfolio P constituted from a set of other securities.

Let r_F be the return of the risk free security and r_P, σ_P^2 the expected return and the risk of the risk portfolio, respectively. Additionally, let us consider $\sigma_F^2 = 0$ for the risk free security and zero correlation between the risk portfolio and the risk free security ($\rho_{FP} = 0$). The investor would be interested to invest a proportion of the available capital w_P to the risk portfolio, and the remaining part $1 - w_P$ to the risk free security. The portfolio risk is given by the following equation:

$$\sigma_{FP}^2 = (1 - w_P)^2 \sigma_F^2 + w_P^2 \sigma_P^2 + 2 w_P (1 - w_P) \rho_{FP}\ \sigma_P\ \sigma_F = w_P^2 \sigma_P^2 \tag{2.28}$$

Therefore, the portfolio risk is defined exclusively as the risk of portfolio P, related to its weighting factor in the portfolio FP. From the above equation we take $w_P = \sigma_{FP}/\sigma_P$. Thus, the portfolio return is:

$$\begin{aligned} E(r_{FP}) &= (1 - w_P) r_F + w_P E(r_P) && \Rightarrow \\ E(r_{FP}) &= (1 - \frac{\sigma_{FP}}{\sigma_P}) r_F + \frac{\sigma_{FP}}{\sigma_P} E(r_P) && \Rightarrow \\ E(r_{FP}) &= r_F + \frac{E(r_P) - r_F}{\sigma_P} \sigma_{RF} \end{aligned} \tag{2.29}$$

The portfolio return is a linear function of risk, resulting to the formulation of a new efficient frontier. As shown in the above diagram, in case that there is a risk free security, the efficient portfolios are placed in the line that intersects the vertical axis at point r_F. Consequently, there is only one optimal portfolio FP, which is determined by the risk portfolio P placed in the tangent of the above equation with the risk portfolios efficient frontier (figure 2.2).

The portfolio P includes m risk securities with weighting factors $w_1, w_2, \ldots, w_m$ and the risk free security with weighting factor w_F, such that $w_1 + w_2 + \cdots + w_m + w_F = 1$. The problem lies to the definition of the proportions of the portfolio FP. The portfolio return and risk are calculated as follows:

$$\begin{aligned} E(r_{FP}) &= w_F r_F + \sum_{i=1}^{m} w_i E(r_i) && \Rightarrow \\ E(r_{FP}) &= \left(1 - \sum_{i=1}^{m} w_i\right) r_F + \sum_{i=1}^{m} w_i E(r_i) && \Rightarrow \\ E(r_{FP}) &= r_F + \sum_{i=1}^{m} w_i [E(r_i) - r_F] \end{aligned} \tag{2.30}$$

$$\sigma_{FP} = \sum_{i=1}^{m} \sum_{j=1}^{m} w_i w_j \sigma_{ij} \tag{2.31}$$

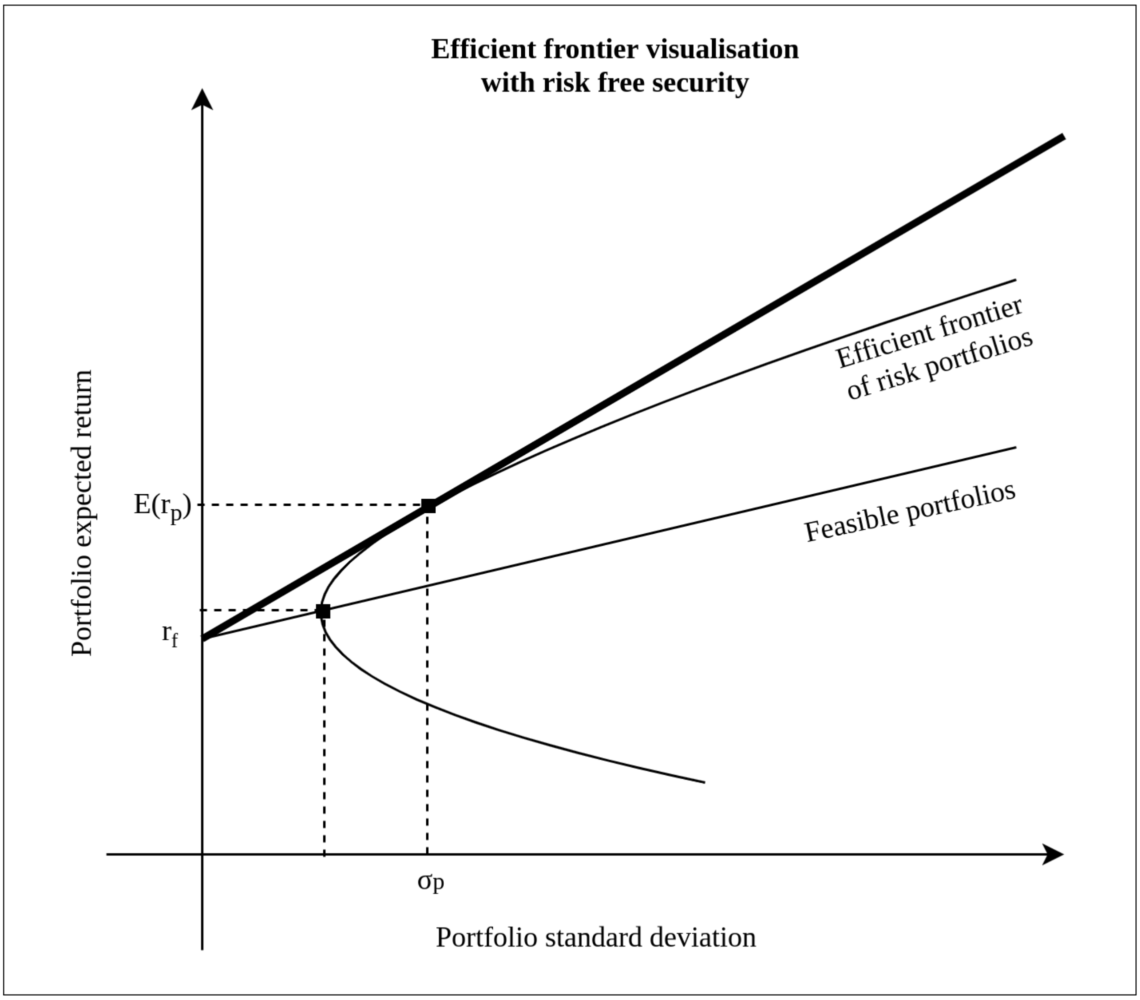


Figure 2.2: Visualisation of the efficient frontier with risk free security

At this point we present the suitable techniques for portfolio optimisation considering two main cases (short sales allowance and short sales restriction).

Short sales allowed

In case that short sales are allowed, the problem is similar to the respective problem without risk free security. Two different proofs are provided. The first proof based on the matrix form of the problem follows the same approach with the previous chapter using the lagrangian multipliers. The second one is a pure algebraic proof, and it is based on the maximisation of the frontier slope.

The following proof, given by Doumpos [25] describes the solution of the initial problem.

Proof. Let r_F represent the return of the risk free security. The purpose of this analysis is the composition of the optimal risk portfolio, given the expected return

R. In vector form the problem is formulated as follows:

$$\begin{array}{rl} \min\limits_{\mathbf{w}} & \sigma_{FP}^2 = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} \\ \text{s.t.} & r_F + (\mathbf{r} - r_F\mathbf{e})^T\mathbf{w} \;\; = \;\; R \\ & \mathbf{w} \in \mathbb{R} \end{array} \tag{2.32}$$

where $\mathbf{e}$ represents the unary column-vector: $\mathbf{e} = (1, 1, \ldots, 1)^T$ and $\mathbf{V}$ is a positive definite matrix.

Let λ be the Lagrange multiplier of the constraint, the following function is formulated:

$$\mathscr{L} = \frac{1}{2}\mathbf{w}^T\mathbf{V}\mathbf{w} + \lambda\left[R - r_F - (\mathbf{r} - r_F\mathbf{e})^T\mathbf{w}\right] \tag{2.33}$$

In order to maximise this function the partial derivatives are set equal to zero:

$$\left\{\begin{array}{l} \dfrac{\partial\mathscr{L}}{\partial\mathbf{w}} = 0 \Rightarrow \;\; \mathbf{V}\mathbf{w} - \lambda(\mathbf{r} - r_F\mathbf{e}) = 0 \\ \dfrac{\partial\mathscr{L}}{\partial\lambda} = 0 \Rightarrow \;\; R - r_F - (\mathbf{r} - r_F\mathbf{e})^T\mathbf{w} = 0 \end{array}\right\} \tag{2.34}$$

The first equation of (2.34) gives us:

$$\mathbf{w} = \lambda\mathbf{V}^{-1}(\mathbf{r} - r_F\mathbf{e}) \tag{2.35}$$

so substituting $\mathbf{w}$ to the second equation gives:

$$R - r_F = \lambda(\mathbf{r} - r_F\mathbf{e})^T\mathbf{V}^{-1}(\mathbf{r} - r_F\mathbf{e}) \Leftrightarrow \lambda = \frac{R - r_F}{(\mathbf{r} - r_F\mathbf{e})^T\mathbf{V}^{-1}(\mathbf{r} - r_F\mathbf{e})} = \frac{R - r_F}{d}$$

Now, substituting this equation to (2.35), $\mathbf{w}$ can be calculated as follows:

$$\mathbf{w} = \frac{R - r_F}{d}\mathbf{V}^{-1}(\mathbf{r} - r_F\mathbf{e}) \tag{2.36}$$

The vector $\mathbf{w}$ determines the risk portfolio composition. The weighting factor of the risk free security in portfolio FP is then calculated as follows:

$$w_F = 1 - \sum_{i=1}^{m} w_i \tag{2.37}$$

The portfolio risk is given by the first equation of (2.34)

$$\begin{aligned} \mathbf{V}\mathbf{w} - \lambda(\mathbf{r} - r_F\mathbf{e} = 0 &\Leftrightarrow \\ \mathbf{w}^T\mathbf{V}\mathbf{w} - \lambda\mathbf{w}^T(\mathbf{r} - r_F\mathbf{e} = 0\mathbf{V}\mathbf{w} &\Leftrightarrow \\ \sigma_{FP}^2 - \lambda[R - r_F] = 0 &\Leftrightarrow \\ \sigma_{FP}^2 = \frac{(R - r_F)^2}{d} & \end{aligned} \tag{2.38}$$

From the above equation we can see that the portfolio return R is a linear function of risk:

$$R = r_F + \sigma_{FP}\sqrt{d} \tag{2.39}$$

□

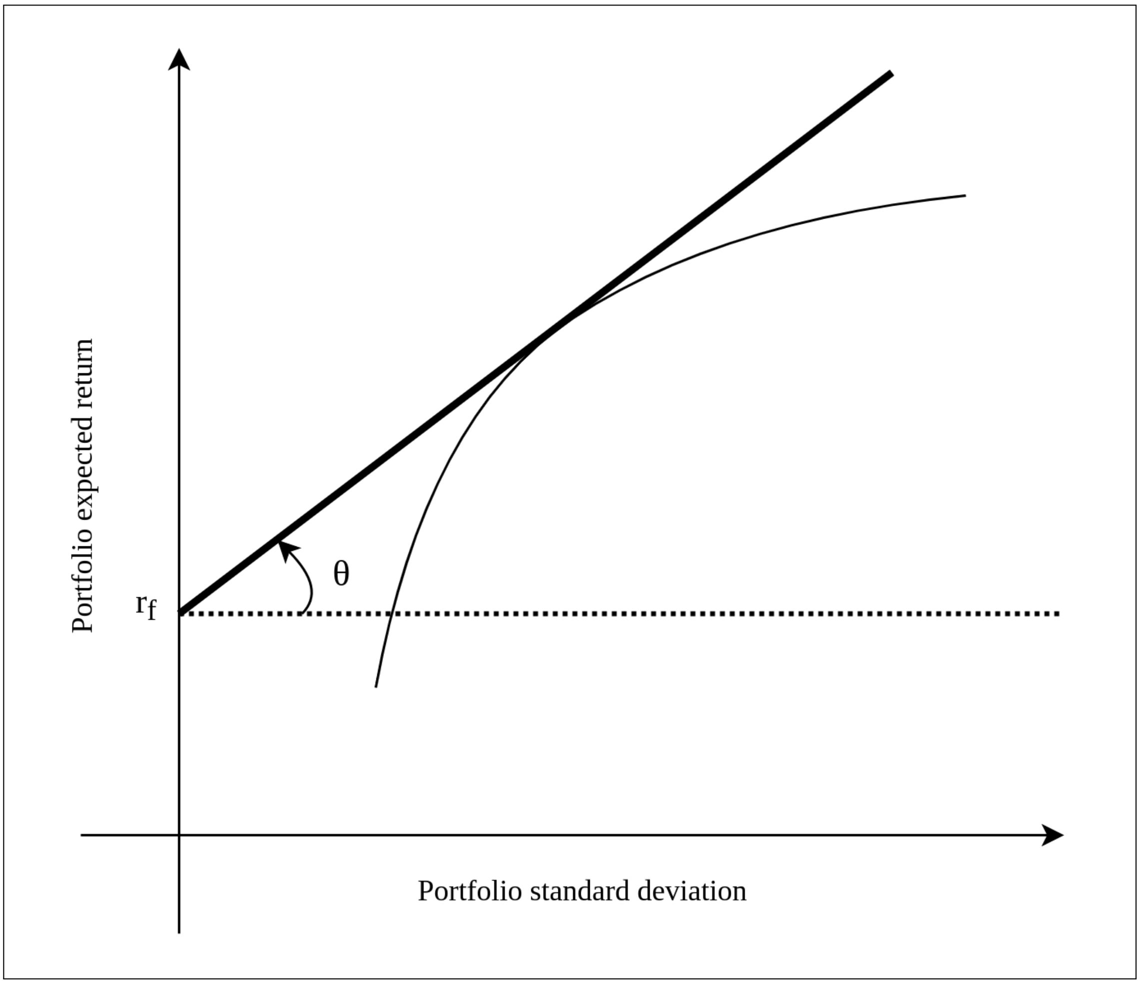


Figure 2.3: Maximisation of the envelope θ for the determination of the efficient frontier given a risk free security

The second proof for this case is based on the maximisation of the slope of the efficient frontier. This proof was given by Xidonas et al. [24]. The slope of the line that relates the risk free security to the risk portfolio is equal to the excess return of the portfolio divided to its volatility (figure 2.3). The excess return of a portfolio represents the difference between the portfolio expected return and the risk free security return. Therefore, the aim is to maximise the slope and the problem is formulated as follows:

$$\begin{aligned} \max_{w} \quad & \theta = \frac{E(r_P) - r_F}{\sigma_P} \\ \text{s.t.} \quad & \sum_{i=1}^{m} w_i \;=\; 1 \\ & w_i \;\in\; \mathbb{R} \qquad i = 1, 2, \ldots, m \end{aligned} \tag{2.40}$$

The problem could be solved like the previous ones, with lagrange multipliers. However, a different approach is followed, as the only restriction of the problem is incorporated in the objective function. Thus, the problem is transformed to a maximisation problem with no restrictions. The proof is given below:

Proof. Given that:

$$r_F = \left(\sum_{i=1}^{m} w_i\right) r_F = \sum_{i=1}^{m}(w_i r_F)$$

the objective function takes the following form:

$$\theta = \frac{\sum_{i=1}^{m} w_i(r_i - r_F)}{\left(\sum_{i=1}^{m} w_i^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,\ j\neq i}^{m} w_i w_j \sigma_{ij}\right)^{1/2}} \tag{2.41}$$

The weighting factors w_k of the invested capital -which maximise the objective function θ can be determined solving the following system of partial derivatives $\partial\theta/\partial w_k$ if they are set equal to zero:

$$\begin{aligned} \frac{\partial \theta}{\partial w_k} = 0 \Rightarrow \\ (r_k - r_F)\left(\sum_{i=1}^{m} w_i^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,\ j\neq i}^{m} w_i w_j \sigma_{ij}\right)^{-1/2} + \left[\sum_{i=1}^{m} w_i(r_i - r_F)\right] \\ \left[-\frac{1}{2}\left(\sum_{i=1}^{m} w_i^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,\ j\neq i}^{m} w_i w_j \sigma_{ij}\right)^{-3/2}\left(2w_k\sigma_k^2 + 2\sum_{j=1,\ j\neq k}^{m} w_j\sigma_{kj}\right)\right] = 0 \Rightarrow \\ (r_k - r_F) - \left(\frac{\sum_{i=1}^{m} w_i(r_i - r_F)}{\sum_{i=1}^{m} w_i^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,\ j\neq i}^{m} w_i w_j \sigma_{ij}}\right)\left(w_k\sigma_k^2 + \sum_{j=1,\ j\neq k}^{m} w_j\sigma_{kj}\right) = 0 \end{aligned} \tag{2.42}$$

Let λ be:

$$\lambda = \frac{\sum_{i=1}^{m} w_i(r_i - r_F)}{\sum_{i=1}^{m} w_i^2\sigma_i^2 + \sum_{i=1}^{m}\sum_{j=1,\ j\neq i}^{m} w_i w_j \sigma_{ij}} \tag{2.43}$$

the last equation gives:

$$\begin{aligned} (r_k - r_F) - \lambda\left(w_k\sigma_k^2 + \sum_{j=1,\ j\neq k}^{m} w_j\sigma_{kj}\right) = 0 \\ r_k - r_F = \lambda w_k\sigma_k^2 + \sum_{j=1,\ j\neq k}^{m} \lambda w_j\sigma_{kj} \end{aligned} \tag{2.44}$$

Let the new variable $Z_\kappa = \lambda w_k$, the last equation gives for every variable i:

$$r_i - r_F = Z_1\sigma_{1i} + Z_2\sigma_{2i} + \cdots + Z_i\sigma_i^2 + \cdots + Z_m\sigma_{mi} \tag{2.45}$$

thus formulating the following linear system:

$$\left\{\begin{array}{l} r_1 - r_F = Z_1\sigma_1^2 + Z_2\sigma_{21} + \cdots + Z_m\sigma_{m1} \\ r_2 - r_F = Z_1\sigma_{12} + Z_2\sigma_2^2 + \cdots + Z_m\sigma_{m2} \\ \;\;\cdots\cdots \\ r_m - r_F = Z_1\sigma_{1m} + Z_2\sigma_{2m} + \cdots + Z_m\sigma_m^2 \end{array}\right\}$$

The weighting factors w_k of the invested capital are obtained solving the above linear system using the following equation:

$$w_k = \frac{Z_k}{\sum_{i=1}^{m} Z_i} \tag{2.46}$$

□

Short sales restricted

In case that short sales are not allowed, the problem is similar to the one with short sales allowed, but an additional restrictrion concerning the weighting factors is added ($w_i \geq 0$). The definition of the problem is presented below:

$$\begin{array}{ll} \max\limits_{w} & \theta = \dfrac{E(r_P) - r_F}{\sigma_P} \\ \text{s.t.} & \sum\limits_{i=1}^{m} w_i \;=\; 1 \\ & w_i \;\geq\; 0 \qquad i = 1, 2, \ldots, m \end{array} \tag{2.47}$$

This is a quadratic programming problem with linear restrictions. The solution of this problem can be given using algorithm based on Kuhn-Tucker conditions. These conditions secure that if a solution is found, then this solution is guaranteed to be optimal. The Kuhn-Tucker conditions are presented as follows:

$$\begin{aligned} \frac{\partial \theta}{\partial w_i} + U_i = 0 \\ w_i U_i = 0 \\ w_i \geq 0 \\ U_i \geq 0 \end{aligned} \tag{2.48}$$

Any solution that satisfies these conditions, is an optimal portfolio of the efficient frontier.

2.6 Conclusion

The preceding analysis has articulated the following conclusions:

- The mean-variance model is based on a quadratic mathematical programming problem which involves a significant amount of calculations due to its complexity. More specifically, the calculation of the covariance matrix becomes very difficult in case that the number of securities is large. The complexity of the algorithm for an input of n securities is $n(n-1)/2$, i.e. $O(n^2)$, thus making the problem non linear. Various alternative measures of risk have been proposed, in order to overcome this problem, such as the *mean absolute deviation* (MAD) which would result in a linear problem.

- The proposed model is based on two criteria (return and risk), thus failing to create a realistic framework of the problem. On the contrary, an integrated approach demands the incorporation of all parameters that affect the security market. Consequently, the portfolio management problem is a multiple-criteria problem, like the majority of decision problems nowadays.

- The conventional approach fails to insert all the components of risk in the methodological framework. This weakness has been cured by other models which embody multidimensional nature of risk, such as the *capital asset pricing model* (CAPM) of Sharpe (1964).

- The classic approach does not take into consideration the decision-maker's profile. The preferences of the decision-maker are of enormous significance in the process of portfolio management. A risk averse investor would probably care about minimising the risk, while an aggresive investor might prefer to maximise the return.

- Finally, the assumptions of the mean-variance approach have been criticised as weak, because the statistical indexes that are used do not have a brief explaination in economic theory. For example, it is said that the assumption which considers that the return of securities follow normal distribution, is unrealistic.

In conclusion, the problem of portfolio management has a multicriterial nature, as it includes a variety of factors. Consequently, the need for integrated methodologies is imperative in order to cure the above-mentioned problems.

Chapter 3

Related Methodologies Overview

3.1 Introduction

The discussion of the portfolio management problem signified the need for new methodological decision support frameworks, in order to overcome the existing problems and cure the inadequacies of the conventional mean-variance model.

In this chapter, there is a short introduction to multiple-criteria decision analysis, as it is the most appropriate field to support the portfolio management decision-making process, according to Xidonas et al. [23].

In the first section there is a presentation of the basic concepts of this scientific field, as well as a general methodology overview, analysing the four phases of decision support. Additionally, an introduction to discrete multicriteria decision support methods is made, presenting and comparing the three basic sectors (multiattribute utility theory, outranking relations theory, preference disaggregation approach).

In the second section the continuous optimisation methods are developed. Beginning with the fundamental concept of linear programming, in the following paragraphs there is a discussion about quadratic and integer programming problems. The following paragraph is about multiobjective programming problems. Afterwards, the methodological framework of goal programming problems is presented, followed by an introduction to genetic algorithms.

All these techniques are involved in some part of the proposed methodology, necessitating the need for a brief introduction before the development of the methodological decision support framework.

3.2 Discrete Multiple-Criteria Decision Analysis

Multiple-criteria decision analysis (MCDA) or Multiple-criteria decision-making (MCDM) belongs to the scientific field of operational research. The main objective of all methodological approaches in the field of multicriteria decision analysis is the development of models that incorporate all the parameters of the problem in order to support the decision-maker in the decision-making process.

3.2.1 Basic concepts and methodology

In 1985, Roy, one of the founders of multicriteria analysis modern theory, presented a general methodological framework for multidimensional decision-making problems [3]. As shown in figure 3.1, the analysis process of multicriteria decision making problems involves four stages among which feedback can be developed.

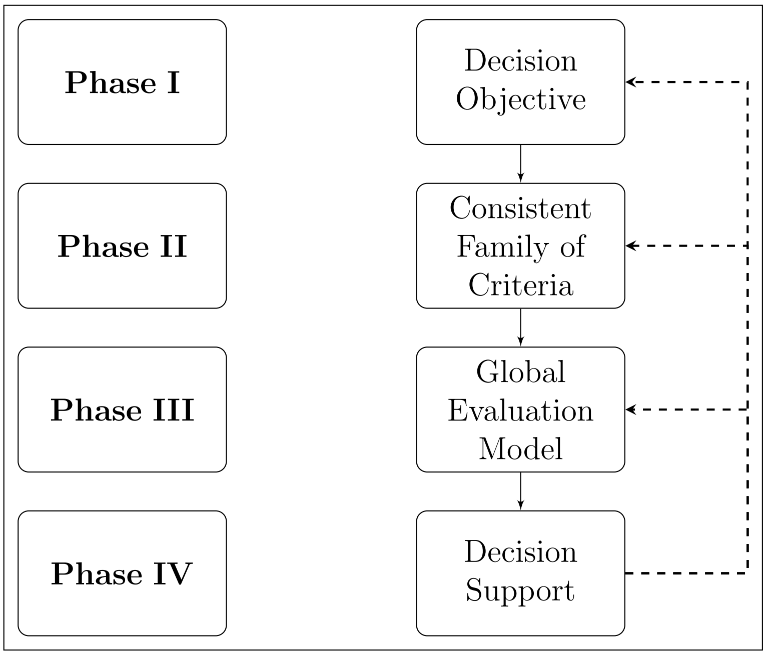


Figure 3.1: Multicriteria decision analysis methodological framework (Roy, 1985)

Phase 1: Decision Objective

In this phase, there are two basic tasks which is necessary to be completed: (a) Strict definition of set A of alternatives or actions of the problem and (b) identification of the decision problematic.

The set A of the alternatives of the problem could be a continuous set or a discrete set. In the case of a continuous problem, the continuous set of solutions is defined by mathematical equations (linear inequalities) as a super-hydride with as many dimensions as the multitude of the decision variables. In the case of a discrete problem, the set of feasible solutions is defined by the exhaustive enumeration of its elements.

The decision problematic determines the way that alternatives should be examined. According to Roy (1985) [3] , there are four main categories of discrete problems (figure 3.2):

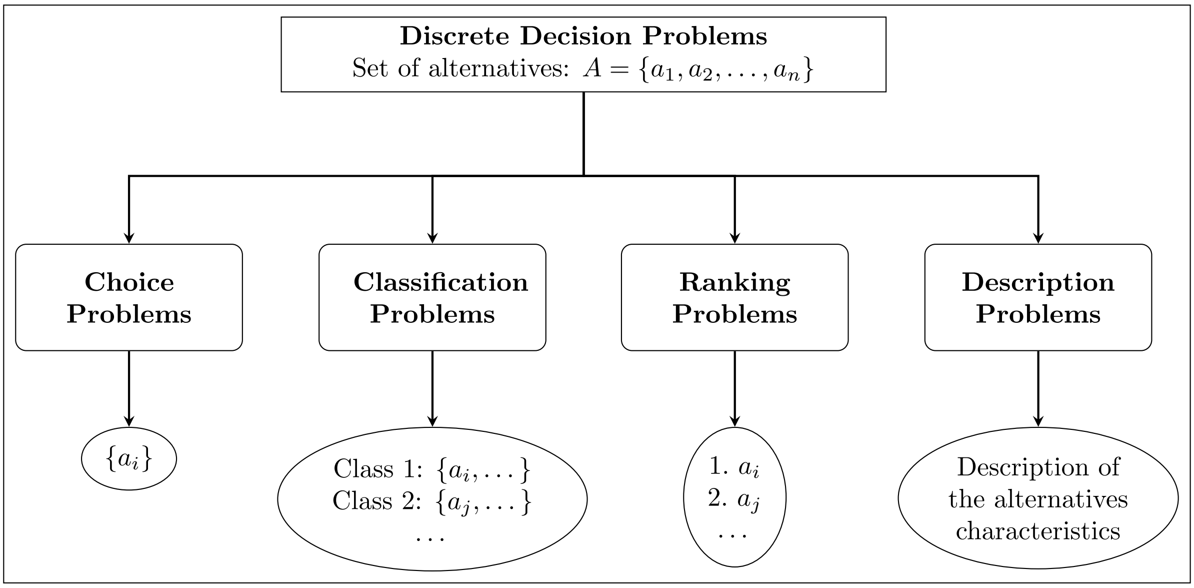


Figure 3.2: Discrete decision problematics

i. Choice problems refer to the situation where the DM must choose the most suitable alternatives.

ii. Classification problems refer to the situation where the alternatives must be classified in predefined classes.

iii. Ranking problems refer to the situation where the alternatives should be ranked in decreasing order.

iv. Description problems refer to the situation where the alternatives are described according to their performance in individual criteria.

Phase 2: Consistent Family of Criteria

Each factor that affects a decision is considered to be a criterion. Formally, a criterion is a monotonic function f which declares the preference of the decision maker, so as for any two alternatives x_i, x_j the following equations apply:

$$f(x_i) > f(x_j) \Leftrightarrow x_i \succ x_j \tag{3.1}$$

$$f(x_i) = f(x_j) \Leftrightarrow x_i \sim x_j \tag{3.2}$$

where the notation $x_i \succ x_j$ declares that alternative x_i is preferable to x_j and the notation $x_i \sim x_j$ declares that there is indifference between the two alternatives.

This procedure results in the configuration of a consistent family of criteria. A set of criteria $F = \{f_1, \dots, f_q\}$ configures a consistent family of criteria, if and only if the following properties are satisfied:

1. **Monotonicity:** A set of criteria is considered monotonic if and only if for any two alternatives x_i, x_j such that $f_k(x_i) > f_k(x_j)$ for any criterion k and $f_l(x_i) = f_l(x_j)$ for any other criterion $l \neq k$, it is concluded that $x_i \succ x_j$.

2. **Exhaustivity:** A set of criteria is considered to be exhaustive if and only if for any two alternatives x_i, x_j such that $f_k(x_i) = f_k(x_j)$ for any criterion k, it is concluded that $x_i \sim x_j$.

3. **Non-redundancy:** A set of criteria is considered to be non-redundant if and only if the removal of any criterion, leads to monotonicity or exhaustivity property violation.

Phase 3: Global Evaluation Model

The global evaluation model is defined as the composition of all the criteria, in order to analyse the problem according to the determined problematic. The global evaluation model can be applied to determine a total evaluation of the alternatives, to explore the solution set (for continuous problems) and to execute pairwise comparisons between all pairs of alternatives.

Phase 4: Decision Support

This phase of the process involves all the activities which help the decision maker understand the results of the application of the model. The role of the consultant is of crucial importance because he must organise the answers in a comprehensible way.

There are three main fields which deal with discrete multiple-criteria decision problems:

- Multiattribute utility theory
- Outranking relation theory
- Preference disaggregation approach

3.2.2 Multiattribute utility theory

Multiattribute utility theory (MAUT) constitutes a generalisation of classical utility theory. From the early stages of multicreteria decision analysis, multiattribute utility theory has been one of its fundamental subfields, supporting both its theoretical and its practical evolution.

Mutliattribute utility theory uses a *value function* or (utility function) $U(g)$ which represents the value system that the decision maker follows. This function has the following expression:

$$U(\mathbf{g}) = U(g_1, g_2, \ldots, g_n) \tag{3.3}$$

where $\mathbf{g}$ is the evaluation criteria vector: $\mathbf{g} = g_1, g_2, \ldots, g_n$.

In general, utility functions are non-linear monotonically increasing functions that meet the following properties:

$$U(\mathbf{g}_x) > U(\mathbf{g}_{x'}) \Leftrightarrow x \succ x'$$

and

$$U(\mathbf{g}_x) = U(\mathbf{g}_{x'}) \Leftrightarrow x \sim x'$$

The most widely known form of the utility function is the additive:

$$U(\mathbf{g}) = p_1 u_1(g_1) + p_2 u_2(g_2) + \cdots + p_n u_n(g_n) \tag{3.4}$$

where u_i are the partial utility functions of the evaluation criteria and p_i are the criteria weighting factors, which should sum to one. Each weighting level implies the trade-off that the decision-maker is willing to pay, in order to succeed unary increasement over the corresponfing criterion.

The additive utility function is based on the important hypothesis of *mutual preferential independence* of the evaluation criteria, which is explained as follows: A subset g' of the set of evalutaion criteria $g' \subset g$, is considered to be *preferential independent* of the rest of the criteria, if and only if the preference of the decision-maker about the alternatives, which differ only in terms of the criteria of g', are not affected by the rest of the criteria. The set of the evaluation criteria is considered to fulfill the assumption of mutual preferential independence, if and only if each subset is preferential independent of the rest of the criteria.

The utility function construction process should be based on the cooperation of the decision-maker himself with an expert analyst. The significance level of the evaluation criteria, as well as the form of the partial utility function must be determined before the construction of the utility function. The determination of the partial utility functions, is based on interactive techniques, such as direct questions to the decision maker, which lead to a detailed understanding of the way the decision-maker evaluates the alternatives in each criterion. The best known technique is called *midpoint value technique* (Keeney and Raiffa, 1993).

3.2.3 Outranking relations theory

The *outranking relations theory* is a special methological multicreteria analysis sector, which emerged at late 1960s with Bernard Roy's study and the presentation of the *ELECTRE* family methods (ELimination Et Choix Traduisant la Realite) (Roy, 1968, 1991, 1996)[2] [5] [4] and has widely spread, especially in Europe, since. It must be noted that the outranking relations theory has its roots social choice theory (Arrow and Raynaud,1986) [9].

Unlike multiattribute utility theory which aims at the development of a utility function, the goal of the outranking relations theory is the development of a methological framework that allows pairwise comparison between alternatives. All the techniques that are based on the outranking relations theory are applied in two basic phases. The first phases includes the development of an outranking relation between the examined alternatives, while in the second phase the outranking relation is exploited for the evaluation of the alternatives in the desired form (ranking, classification, choice).

The outranking relation **S** is a bilateral relation defined in the set of alternatives, such that:

$$x\mathbf{S}x' \Leftrightarrow \text{alternative } x \text{ is at least as good as alternative } x' \qquad (3.5)$$

The idea of outranking relation is that the comparison of two alternatives x and x' is based on the power of both *positive indications*, which support the fact that alternative i is better than alternative j and *negative indications*, which support the opposite fact. In case that the power of positive indications is significant and the power of negative indications is insignificant, we can assume that there is an outranking relation $x\mathbf{S}x'$ between alternatives x and x'.

In fact, the outranking relations theory differs from the multiattribute utility theory in two major points:

- The outranking relation is not transitive. In utility theory the evaluation of the alternatives with the utility function maintains the transitive property. On the contrary, the development and use of outranking relations allows the representation of cases where, while the alterative x_1 is preferable to x_2, and x_2 is preferable to x_3, finally x_1 is neither preferable nor indifferent to x_3.

- The outranking relation is not complete. The completeness property refers to the complete evaluation and ranking of all the alternatives. The multiattribute utility theory leads to a complete evaluation of the alternatives, developing appropriate utility functions. On the other side, the outranking relations theory does not necessarily demand the decision-maker's preferences to carry the transitive property, thus a complete evaluation is often impossible. The non-completeness property is very important due to the fact that the complete ranking of the alternatives is unrealistic in a variety of problems.

Therefore, the *incomparability property* is a property of outranking relations theory, that makes it extremely useful for this category of problems. The information provided by the decision-maker is important for the development of the outranking relation. This information is quite different, depending on the particular method used. However, in the majority of cases it is about (a)the significance of the evaluation criteria (weighting factors) and (b) the preference, indifference and veto thresholds. These thresholds contribute to the development of a fuzzy outranking relation, where there is partial preference or even indifference among the alternatives.

3.2.4 Preference disaggregation approach

The preference disaggregation approach (Jacquet-Lagreze and Siskos, 1982, 2001) [13] [14] invlolves the development of a methological framework which can be used for the analysis of decisions made by the decision-maker, in order to determine the appropriate criteria synthesis model that meets the value system and the preferences of the decision-maker.

The preference disaggregation approach follows a reverse process compared to the multiattribute utility theory and the outranking relations theory. This approach considers that the decision-maker (consciously or not) follows a value system which results in the decisions he makes. It tries to detect the way that decisions are made and finally reproduce a similar decision-making model. In order to manage to imitate the decision-maker, this method requires a training sample consisting of: (a) a set of decision made by the decision-maker, (b) the evaluation of a set of hypothetical actions and (c) the evaluation of a subset of the examined alternatives.

In Table 3.1, there is an overview of the methods of each field, according to Xidonas(2009):

Multiattribute Utility	Outranking Relation	Preference Disaggregation
AHP	ELECTRE	UTA
TOPSIS	QUALIFLEX	UTASTAR
MAUT	PROMETHEE	UTADIS
MCBETH	ORESTE	MHDIS
ANP	REGIME	
	EVAMIX	
	MELCHIOR	
	TACTIC	
	PRAGMA	
	MAPPAC	
	ARGUS	
	IDRA	
	PACMAN	

Table 3.1: Overview of discrete MCDA methods

3.3 Multiobjective Mathematical Programming

3.3.1 Introduction

A problem that requires to choose the best solution from a set of feasible solutions is called *optimisation problem*. These problems are cured by a scientific field which is called *mathematical optimisation* (or *mathematical programming*). There are two main categories of optimisation problems: Discrete optimisation refers to problems with discrete variables and the solution is one or more elements of the feasible set, which is a countable set of possible solutions. Continuous optimisation refers to problems with continuous variables and the solution requires to optimise a continuous function subject to one or more equality or inequality constraints. The concept of discrete optimisation was discussed in the previous section. In this section, a brief introduction to continuous optimisation problems will be made.

The standard form of a continuous optimisation problem is the following:

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0, \quad i = 1, \ldots, m \\ & h_j(x) = 0, \quad j = 1, \ldots, n \end{aligned} \tag{3.6}$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function which should be minimised over a vector x, $g_i(x) \leq 0$ are m inequality constraints and $h_j(x) = 0$ are n equality constraints, where $m,\ n \geq 0$. In the above definition, the objective function should be minimised by convention. In case of maximisation problem, the objective function should be negated, thus transforming the problem to minimisation problem.

In the following paragraphs, some of the major subfields of continuous optimisation -which will be used in the methodological framework- will be discussed.

Linear programming

Linear programming (LP) (Dantzig, 1998) [8] is a continuous mathematical optimisation method which is used in a category of optimisation problems, when all the contraints are strictly expressed with linear equations. More specifically, this technique can be used if the objective function of the problem is linear and the requirements of the problem are linear equality or inequality constraints. These linear constraints form a convex polyhedron which is called feasible region of the problem. The linear function is defined on this polyhedron. A linear programming algorithm finds the point where this function has the smallest value (for minimisation problem). Some of the most famous applications of linear programming are in the fields of operational reasearch, enginnering, scheduling and transportation.

Let x represent a vector of decision variables, b, c represent column vectors of known constants and A represent a matrix of known constants. The linear problem canonical form is expressed as follows:

$$\begin{array}{rrcl} \min\limits_{x} & \mathbf{c}^{\mathrm{T}}\mathbf{x} & & \\ \text{s.t.} & A\mathbf{x} & \leq & \mathbf{b} \\ & \mathbf{x} & \geq & \mathbf{0} \end{array} \tag{3.7}$$

The inequalities $Ax \leq b$ and $x \geq 0$ are linear constraints which determine a convex polytope.

A more intuitive definition of a linear programming problem is the standard form, which includes the description of the objective function, the decision variables and the problem constraints. There are two categories of linear programming algorithms. The first category consists of exchange algorithms, such as the simplex algorithm and the criss-cross algorithm. The simplex algorithm was introduced by Dantzig in 1947 and it is one of the most efficient LP algorithms. The algorithm detects a feasible solution at a random vertex of the polyhedron and then walks along the other vertices until the optimal solution is found. The second category consists of interior point algorithms, such as the ellipsoid algorithm, the projective algorithm of Karmarkar etc.

Quadratic programming

Quadratic programming (QP) is a continuous mathematical optimisation method which is used for linear constrained quadratic optimisation problems. More specifically, the objective function is a quadratic function and the problem restrictions are formed as linear equations and inequations. QP is a specific type of nonlinear optimisation.

Let x represent a vector of n decision variables, c a n-dimensional vector, b a m-dimensional vector, Q a $n \times n$ symmetric real matrix and A a $m \times n$ real vector. The quadratic programming problem with n variables and m constraints is formulated as follows:

$$\begin{array}{rrcl} \min\limits_{x} & \frac{1}{2}\mathbf{x}^{\mathrm{T}}Q\mathbf{x} + \mathbf{c}^{\mathrm{T}}\mathbf{x} & & \\ \text{s.t.} & A\mathbf{x} & \leq & \mathbf{b} \end{array} \tag{3.8}$$

The solution of quadratic programming problems is particularly simple when Q is positive definite and there are only equality constraints. The solution is produced using Lagrange multipliers and seeking the extremum of the Lagrangian. The portfolio optimisation problem, which was discussed in section 2.2.4, belongs to this category. Therefore given the quadratic problem

$$\begin{array}{rrcl} \min\limits_{x} & \frac{1}{2}\mathbf{x}^{\mathrm{T}}Q\mathbf{x} + \mathbf{c}^{\mathrm{T}}\mathbf{x} & & \\ \text{s.t.} & A\mathbf{x} & = & \mathbf{b} \end{array} \tag{3.9}$$

if λ is a set of Lagrange multipliers, the solution is given by the linear system:

$$\begin{bmatrix} Q & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \lambda \end{bmatrix} = \begin{bmatrix} -\mathbf{c} \\ \mathbf{b} \end{bmatrix} \tag{3.10}$$

If Q is a positive definite matrix, the appropriate algorithm is the ellipsoid method which provides the solution in polynomial time. Otherwise, the problem is NP-hard, which means that there is not algorithm with polynomial complexity for the problem.

Integer programming

Integer programming (IP) is a mathematical optimization method for some particular problems where all the variables are restricted to be integers. In the special case that the objective function and a part of the constraints are linear expressions but the decision variables are integer variables, the appropriate method is called integer linear programming (ILP). Particularly, in case that even some of the decision variables are continuous, the method is called mixed-integer programming (MIP). Finally, in case that both some of the decision variables are continuous and the objective function or some of the constraints are linear functions, then the problem belongs a particular subcategory called mixed-integer linear programming (MILP). MILP problems will be part of the proposed methodological framework.

The canonical form of an ILP problem is expressed as:

$$\begin{aligned} \min \quad & \mathbf{c}^{\mathrm{T}}\mathbf{x} \\ \text{s.t.} \quad & A\mathbf{x} \leq \mathbf{b}, \\ & \mathbf{x} \geq \mathbf{0}, \\ \text{and} \quad & \mathbf{x} \in \mathbb{Z}^n, \end{aligned} \tag{3.11}$$

where $\mathbf{c}, \mathbf{b}$ are vectors and A is a matrix, under the constraint that all entries are integers.

There are two subcategories of integer programming depending on the range of the decision variables. The first subcategory involves all the problems which demand a decision to be taken. In these problems the integer variables represent decisions with possible answers 'yes' or 'no', which are translated as 1 or 0. Therefore, problems in which integer variables are restricted in the range $\{0, 1\}$ are called belong to *zero-one linear programming*. The second subcategory involves problems in which the integer variables represent discrete quantities, p.e. number of discrete pieces of a product. Such problems frequently appear in production planning, scheduling etc.

In the last decades, a variety of algorithms have been developed, in order to find the optimal solution of an IP problem. The most naive approach is called LP relaxation, recommending to remove all integer constraints, solve the relaxed LP problem and then round the solution. However, this technique may find a non-optimal solution or even a non-feasible one if any constraints are violated during the round process. The optimal solution can be found with *cutting plane methods* and *branch and bound* techniques. Finally, another approach to detect an approximate solution to ILP problems is the field of heuristic methods, which do not guarantee an optimal solution but offer better complexity.

3.3.2 Goal programming

Goal programming (GP) (Charnes and Cooper, 1955, 1961) [19] [21] is a multiobjective optimization technique which extends the concept of linear programming in order to solve problems with multiple conflicting objective functions. The goal programming methodology is quite simple, as each objective function is assigned a goal or target value to be achieved, according to the DM's requirements. Any deviation from this target value is punished with a penalty value. Finally, the weighted sum of all penalty values must be minimised. This technique is useful for the following purposes: (a) The determination of the degree that each goal is fulfilled given the available resources, (b) the estimation of the required resources in order to achieve a predefined goal and (c) the computation of the best feasible solution under a varying amount of resources and goal priorities.

Let $f_1(x), f_2(x), \ldots, f_k(x)$ be a set of k objective function and $x \in X$, where X is the feasibleset of the decision vectors. Let us introduce the *deviational variables* (or slack variables) d_i^-, d_i^+ which represent the amount by which each goal deviates from the target value. More specifically, the vector d^- represents the amount by which each goal's target value is *underachieved*, and vector d^+ represents the amount by which each goal's target is overachived. Finally, let g_i represent the target value of the i_{th} objective function. The problem is formulated as follows:

$$\begin{array}{rrcll} \min\limits_{d^+,d^-} & \sum\limits_{i=1}^{k} \frac{w_i^+ d_i^+}{g_i} + \frac{w_i^- d_i^-}{g_i} & & & \\ \text{s.t.} & f_1(x) + d_1^- - d_1^+ & = & g_1 & \\ & f_2(x) + d_2^- - d_2^+ & = & g_2 & \\ & & \cdots & & \\ & f_k(x) + d_k^- - d_k^+ & = & g_k & \\ & d_i^-, d_i^+ & \geq & 0 & \forall i \in \{1, 2, \ldots, k\} \end{array} \tag{3.12}$$

where w_i^+ represents the weighting factor of the overachievement penalty of the i_{th} objective function and w_i^- represents the weighting factor of the underachievement penalty of the i_{th} objective function.

In case that underachievement of a particular goal is undesirable a greater weighting factor w^- is assigned (i.e. $w^- = 1$), else if underachievement is desirable or neutral the weighting factor is set equal to zero ($w^- = 0$). Accordingly, if overachievement of a particular goal is undesirable a greater weighting factor w^+ is assigned (i.e. $w^+ = 1$), else if overachievement is desirable or neutral the weighting factor is set equal to zero ($w^+ = 0$).

The emerging problem is a linear programming minimisation problem which can be easily solved with the methods mentioned above, in order to minimise deviational variables and, subsequently approach the target value of each goal.

3.3.3 Multi-objective programming

Multi-objective optimisation is a sector of MCDA used for mathematical optimisation problems that require more than one objective function to be optimized simultaneously.

More specifically, multiobjective linear programming (Steuer, 1986) [18] is an extension of linear programming in case that there are multiple objective functions $f_i(\mathbf{x}) = \mathbf{c}_i^T\mathbf{x},\ i = 1, 2, \ldots, k$. The problem is formulated as follows:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_n(\mathbf{x})\} \\ s.t. \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0} \end{aligned} \tag{3.13}$$

In multiobjective programming, there is not an optimal solution because of the existance of many objective functions, as it is infeasible to optimise all objective functions simultaneously. Subsequently, the concept of the optimal solution is replaced with the concept of an *efficient (or Pareto optimal) solution*, based on dominance theory. In the following paragraphs, the most important definitions of multiattribute theory are introduced (Doumpos, 2009) [26].

Any solution x which satisfies the restrictions of the problem is called *feasible solution*. The set of all feasible solutions is called *feasible set*.

A feasible solution $\mathbf{x}$ is called *Pareto dominant* to another feasible solution $\mathbf{x}'$ if and only if: (a) $f_i(\mathbf{x}) \leq f_i(\mathbf{x}')\ \forall i \in \{1, 2, \ldots, k\}$ and (b) $f_j(\mathbf{x}), f_j(\mathbf{x}')$ for at least one $i \in \{1, 2, \ldots\}$.

A solution $\mathbf{x}*$ is called *Pareto optimal*, if and only if there is no other feasible solution that dominates it. The set consinsting of all Pareto optimal solutions is called *Pareto frontier* or *Pareto boundary*.

Finally, a solution x is called *weakly optimal* if and only if ther is no other feasible solution such that $f_i(\mathbf{x}') \geq f_i(\mathbf{x})\ \forall i$.

3.3.4 Genetic Algorithms

Evolutionary computation and algorithms

Evolutionary computation is a scientific subfield of artificial intelligence which is involved with a variety of algorithms for global optimisation. These algorithms are polulation-based trial and error solvers which incorporate metaheuristic or stochastic optimisation characteristics. In biological terms, a population of feasible solutions is subjected to a natural selection process, resulting in a gradual evolution , optimising the *fitness function* of the problem.

Evolutionary algorithms are based on the collective learning process within a population of individuals, each of which represents a search point in the space of potential solutions to a given problem (Back and Schwefel, 1993) . The population is arbitrarily initialized, and it evolves toward better and better regions of the search space with randomized processes of selection, mutation, and recombination. The environment which is called *fitness value* delivers quality information about the search points. The selection process favors those individuals of higher fitness to reproduce more often than those of lower fitness. The recombination mechanism allows the mixing of parental information while passing it to their descendants, and mutation introduces innovation into the population.

Genetic algorithms (GA) are the most known evolutionary algorithms. The concept of general adaptive processes, concentrating on the idea of a system receiving sensory inputs from the environment by binary detectors was initially introduced by J. Holland in 1962. As a result, structures in the search space were modified by operators selected by an adaptive plan, judging the quality of previous trials with an evaluation metric. In a genetic algorithm, a population of individuals is evolved toward better solutions. Each solution has a set of characteristics, called *chromosomes* which can be mutated and altered.

Algorithm 1: Genetic Algorithm Standard Form

```
input: popSize (Size of Population), elit (rate of elitism), mut (rate of
  mutations), maxIter (maximum number of iterations);
create popSize random feasible solutions
save created solutions in population aray
while ! terminal condition do
    for i in range(maxIter) do
        select the best popsize × elit solutions from population
        save them in popul2
        for j in range(crossover) do
            select two random solutions from population
            generate and save new solutions to popul3
        end
        for j in range(crossover) do
            select a solution from popul3 and mutate with ratemut
            if newSolution.isFeasible() then
                Update popul3 with new solution
            end
        end
    end
    update population = pop2 + pop3
end
Result: the best solution in population
```

The algorithm starts from a population of randomly generated solutions, called *individuals*. The process continues iteratively, with the population in each iteration called *generation*. In each generation, the fitness function is evaluated for every individual in the population. The fitness function should represent the objective function of the optimisation problem. Individuals with the best fitting score are selected, while the least desired individuals are rejected. Thus, each individual is modified and a new generation of individuals is created. The new generation is used as input to the next iteration of the process. The algorithm stops in two cases: firstly, if an optimal solution is found subject to the fitness function and secondly, if the maximum number of iteration has been conducted.

The selected genetic algorithm that will be used in the proposed methodology is called differential evolution. Differential evolution (Storn, Price) is a stochastic population based method which is used in order to find the global minimum of a multivariate function. Every time that it passes through the population, the algorithm mutates each candidate solution by mixing it with other candidate solutions in order to create a trial candidate. By default the best solution vector is updated continuously within a single iteration. This is a slight modification of the original differential evolution algorithm, which can lead to faster convergence as trial vectors can immediately benefit from improved solutions.

The genetic algorithms general methodology is presented in figure 3.3

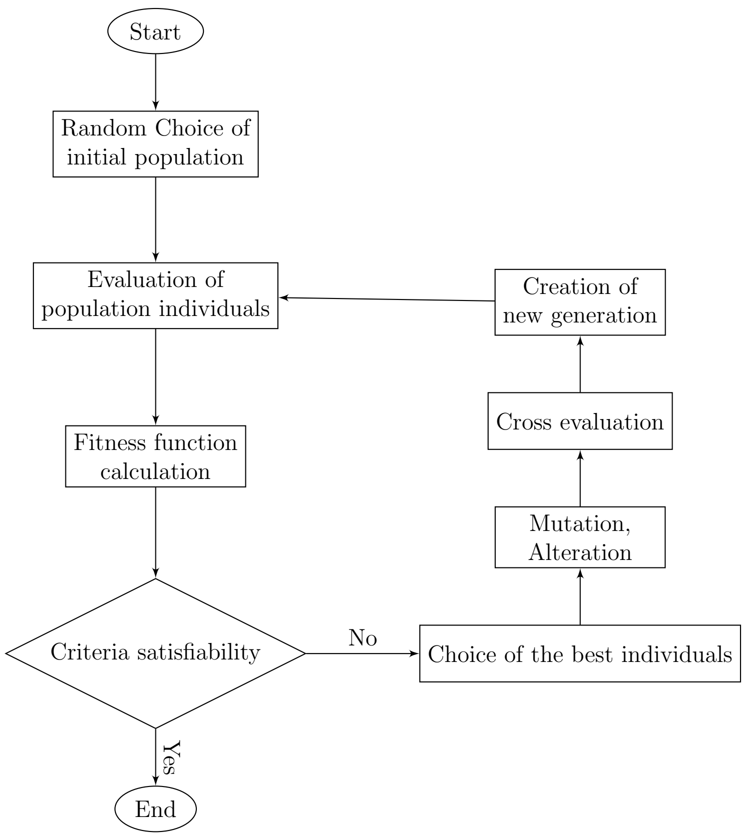


Figure 3.3: Genetic algorithms methodology

3.4 Conclusion

The overview of the related multiobjective methodologies provides the followign concluding points:

- A variety of multiple-criteria decision support methods have been developed, which could be applied to a variety of discrete and continuous decision problems. These techniques seem capable of dealing with many conflicting criteria, which is very useful in the portfolio management problem, having discussed its multivariate nature .
- Some multiple-criteria methods (quadratic and multiobjective programming, complex genetic algorithms etc.) have large calculation load. Therefore, in case that there are many alternatives the solution of some problems is difficult or even unfeasible, making necessary to find an approximate solution or a subset of all the feasible solutions.
- The scientific field of portfolio management has great improvement margin. The need of integrated methodologies and decision support systems are of great necessity, in order to improve the existing methodologies.

Chapter 4

Proposed Methodology

4.1 Introduction

As already mentioned, the purpose of the thesis is the development of an integrated methodological framework for portfolio management. The process of portfolio management is a very complex problem, as it consists of two phases which demand a variety of decisions. The first phase focuses on portfolio selection, i.e. the selection of the strongest investment opportunities. The second phase includes portfolio optimization, the determination of the most efficient allocation of the available capital to the selected securities in order to maximize return.

In this chapter, we present the proposed methodology in the context of the thesis. The following analysis is based on Xidonas et al. scientific work presented on Xidonas thesis project [27] [15], as well as the book Multicriteria Portfolio Management [1]. The ultimate goal is the effective management of security portfolios, which constitute one of the most risky market investments. The proposed methodology aspires to combine existing knowledge with a set of theoritical and practical innovations. In the first phase, four multiple-criteria decision making methods are deployed in order to rank the available securities and detect the best investment opportunities. After the security selection process, the most important financial statistical indexes are calculated based on historical data and, subsequently, a variety of portfolio optimisation models are suggested. The basis of these models is the typical mean-variance approach which remains the basic portfolio optimisation method for more than 60 years. Finally, the selected portfolios are evaluated and compared in order to select the most appropriate portfolio according to the profile of the DM.

This chapter has the following structure: In section 3.2 an overview of the proposed methodology is presented. The two phases of the process are explicitly analysed in sections 3.3 and 3.4. Finally, in section 3.5 there is a conclusion of the main points of the chapter.

4.2 Methodology Overview

In the current section an overview of the proposed methodological framework will be provided, as an introduction to the various methods which are discussed in the following sections. The methodological framework consists of two main phases, (a) portfolio selection and (b)portfolio optimisation as indicated in diagram 4.1.

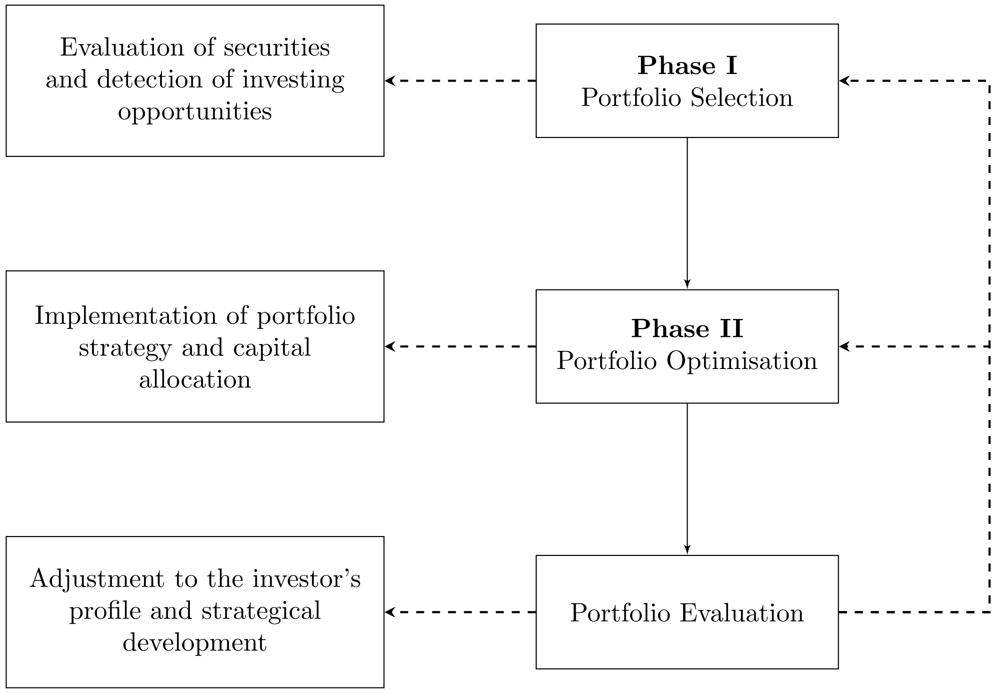


Figure 4.1: Portfolio management methodology phases

The aim of the following overview is to unify all the individual methodologies in a structured and compact framework. As shown in diagram 4.1, the two main phases are not independent from each other. On the contrary, they are interdependent as there is a strong connection and communication among them. Additionally, the whole methodology should be applied in communication with the decision-maker, as it is necessary to interact with the model importing his preferences during the process.

An extensive diagram of the proposed methodological framework is presented in diagram 4.2.

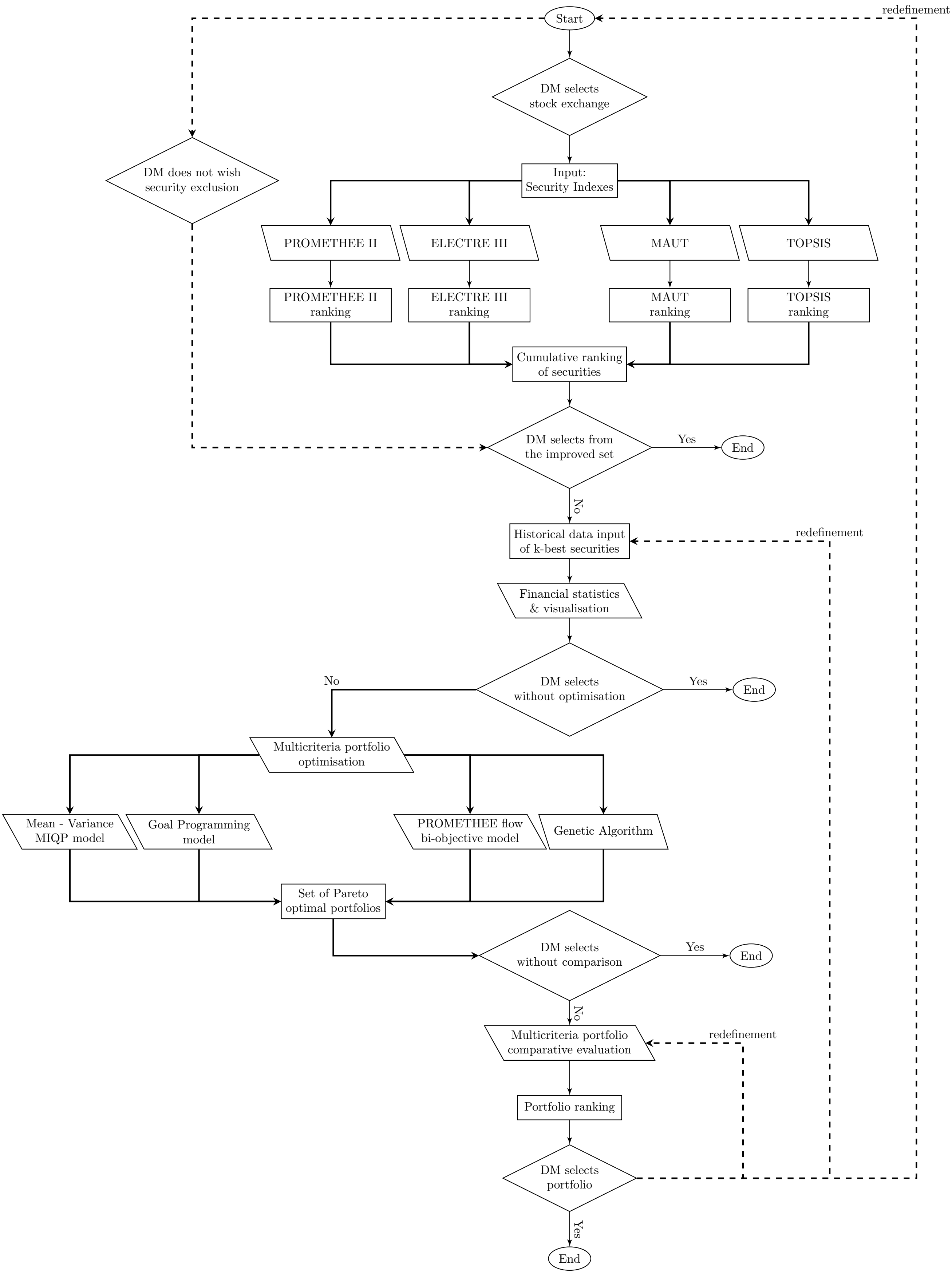


Figure 4.2: Methodological framework extensive presentation

Phase I: Multicriteria Portfolio Selection

The first phase is concerned with the portfolio selection problem, i.e. the construction of a kurn of securities which are considered to be investment opportunities. The decision-maker must select the industrial sector and the stock exchange that he wants to get involved with, resulting in a pool of securities which constitute the problem alternatives. The problem of security selection is healed with multiple-criteria decision analysis (MCDA). More specifically, four ranking MCDA methods are applied to the alternatives, under a variety of financial indexes which serve as the criteria of the multiobjective optimisation problem. Each method provides a ranking of the securities and finally the cumulative ranking of the securities can be calculated as the weighted average of the four individual rankings. After the whole process, the desision-maker can either construct a portfolio with the $k-$best ranked securities, or redefine the problem in case that the result is unsatisfaying.

Phase II: Multiobjective Portfolio Optimisation

The second phase is concerned with the problem of portfolio optimisation. Initially, some of the most major financial indexes are calculated based on the historical data of the improved set of securities. These indexes are significant for two reasons: Firstly, they serve as an additional decision support instrument and secondly they are necessary for the covariance matrix calculation as it will be discussed in the following paragraphs. The original optimisation problem was a bi-criteria problem where the expected return is maximized and the risk is minimized (Markowitz 1952). The methodological framework of this thesis proposes a series of models to approach the problem. Firstly, a bi-objective integer programming problem is formulated based on the mean - variance model, where additional integer constraints are imposed in order to control the weighting factor of each security. Secondly, another bi-objective optimisation approach is introduced where the net flow of PROMETHEE method should be maximized and portfolio beta index should be minimised. Additionally, a goal programming methodology is introduced. Finally, an implementation of a genetic algorithm for portfolio optimisation is presented.

4.3 Phase I: Multicriteria Portfolio Selection

Multiple criteria decision analysis (MCDA) methods are widely used for the study of a wide variety of financial problems. More specifically, the problem of security selection involves all the features of MCDA, such as alternatives, evaluation criteria and objective functions, rendering it one of the most suitable approaches. In this phase of the process, the decision- maker, has to evaluate and select the securities that are available as investment opportunities. This step is necessary because of the vast amount of securities traded in international stock markets. Consequently, the portfolio should consist of a limited number of those securities, excluding securities which have undesirable characteristics.

Before presenting a detailed description of the problem, it is necessary to emphasize the following characteristics of the methodology:

- Firstly, the process of security evaluation is based on specific financial indexes, after an extensive study of the existing literature. These indexes constitute the evaluation criteria of the analysis and will be thoroughly analysed in the following paragraph.
- Secondly, the companies should be categorized into predefined classes before the evaluation is applied, depending on their activity and the industrial sector they belong to. The necessity of this step derives from the fact that the comparison of financial indexes among companies of different industrial sectors would be a contentious assumption.

The methodological framework of the first phase is depicted in diagram 4.3

In the following paragraphs, a detailed description of the portfolio selection problem is given:

Problem Definition

Let $A = \{a_1, ..., a_n\}$ be a set of n alternatives and let $F = \{f_1, ..., f_q\}$ be a consistent family of q criteria. Without loss of generality, we make the assumption that the above criteria should be maximized. Therefore, let us consider the following multicriteria problem:

$$\max\{f_1(a), f_2(a), ..., f_q(a)|a \in A\} \tag{4.1}$$

The input of the above problem is imported in a 2-dimensional table, containing $n \times q$ evaluations, which is called evaluation matrix. Each row corresponds to an alternative action and each column corresponds to an evaluation criterion. The element of the i_{th} row and the k_{th} column describes the performance of alternative

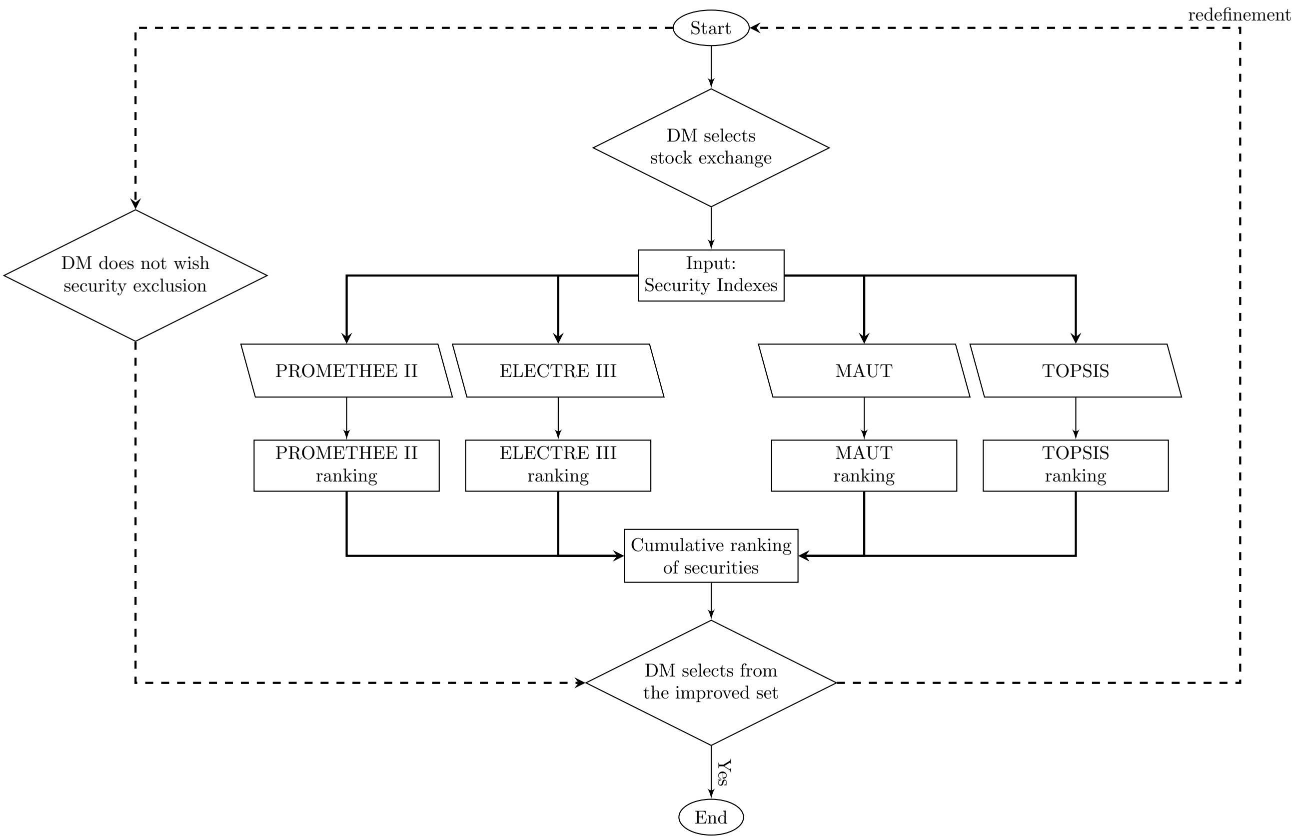


Figure 4.3: Methodological framework for Phase I

a_i in criterion f_k. The evaluation matrix of a multiple criteria decision problem is presented in table 4.1

	$f_1(\cdot)$	$f_2(\cdot)$	$\cdots$	$f_j(\cdot)$	$\cdots$	$f_q(\cdot)$
a_1	$f_1(a_1)$	$f_2(a_1)$	$\cdots$	$f_j(a_1)$	$\cdots$	$f_q(a_1)$
a_2	$f_1(a_2)$	$f_2(a_2)$	$\cdots$	$f_j(a_2)$	$\cdots$	$f_q(a_2)$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
a_i	$f_1(a_i)$	$f_2(a_i)$	$\cdots$	$f_j(a_i)$	$\cdots$	$f_q(a_i)$
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
a_n	$f_1(a_n)$	$f_2(a_n)$	$\cdots$	$f_j(a_n)$	$\cdots$	$f_q(a_n)$

Table 4.1: MCDA problem evaluation table

Problem alternatives

As already mentioned, the set of alternatives $A = \{a_1, ..., a_n\}$ consists of the securities of a specific stock exchange and a specific industrial sector. In the beginning of the process the decision maker selects the industrial sector, as well as the stock exchange. These two components constitute the environment of the study. The set of securities have been classified in 11 classes representing the main industrial sectors are presented in table 4.2.

Class	Industrial sector
A	Basic materials
B	Capital goods
C	Consumer cyclical
D	Consumer non-cyclical
E	Energy
F	Financial
G	Healthcare
H	Services
I	Technology
J	Transportation
K	Utilities

Table 4.2: Classification of industrial sectors

Problem criteria

The evaluation process of securities is based on a set of suitable financial criteria, which depend on the accounting and economic plans of the companies, as well as on experts' analysis. In this paragraph, an extensive presentation of the criteria is provided (table).

	Criteria	Utility	Units
A	Price-to-Earnings Ratio	Minimisation	Percentage
B	Earnings per share	Maximisation	Percentage
C	Revenue	Maximisation	Dollars
D	Beta	Minimisation	Fraction
E	Dividend Yield	Maximisation	Percentage
F	Monthly technical recommendation	Maximisation	Rank
G	Year-to-date performance	Maximisation	Percentage
H	1 year performance	Maximisation	Percentage

Table 4.3: Problem criteria

Offset and thresholds assignment

The determination of the offset of the criteria is a matter of great significance for the efficiency of multicriteria techniques. The main methodologies for offset determination are: (a) the direct weighting system (Hokkanen - Salminen, 1997), (b) the Mousseau system (Mousseau, 1995), (c) the pack of cards technique) (Simos, 1990) and (d) the resistance to change grid method (Rogers and Bruen, 1998). The process of offset allocation must be developed with the assistance of the decision-maker, because his profile and his preferences among the significance of conflicting criteria must be considered during the decision-support model.

Some of the MCDA methods that are used in the proposed methodology include some additional thresholds. In the proposed techniques there are three types of thresholds: (a) *preference threshold*, (b) *indifference threshold* and (c) *veto threshold*. Preference threshold implies that an alternative is totaly preferable to anothe. Indifference threshold signifies that two alternatives are almost equally prefered. Finally, veto threshold represents the threshold that renders a dominated alternative eliminated from the selection process. Threshold determination is a quite complicated process which should be executed in communication with financial experts.

4.3.1 ELECTRE III

The ELECTRE family in MCDA problems is based on the concept of outranking relationship. An alternative a_1 outranks a_2 if and only if there is sufficient evidence to believe that a_1 is better than a_2 or at least a_1 is as good as a_2. More specifically, ELECTRE III method is used for ranking problems, using a structured procedure to calculate the the outranking relationship between each pair of alternatives. It includes a preference threshold, an indifference threshold and a veto threshold.

Let $q(f_i)$ and $p(f_i)$ represent the indifference and preference thresholds for each criterion $f_i,\ i = 1, ..., q$, respectively, and let P denote a strong preference, Q denote a weak preference and I denote indiferrence between a_1 and a_2 for criterion k. If $f_k(a_i) \geq f_k(a_j)$, then:

$$f_k(a_i) > f_k(a_j) + p(f_k) \Leftrightarrow a_1 P a_2$$

$$f_k(a_j) + q(f_k) < f_k(a_i) < f_k(a_j) + p(f_k) \Leftrightarrow a_1 Q a_2$$

$$f_k(a_j) < f_k(a_i) < f_k(a_j) + q(f_k) \Leftrightarrow a_1 I a_2$$

The algorithm of the ELECTRE III method is presented below:

Step 1: Outranking Degree The outranking degree $C_k(a_i, a_j)$, $(0 \leq C_k(a_i, a_j) \leq 1)$ of the alternative a_i and the alternative a_j for criterion f_k is calculated according to the preference definitions (linear interpolation):

$$C_k(a_i, a_j) = \begin{cases} 0 & \text{if} \quad f_k(a_j) - f_k(a_i) > p(f_k) \\ 1 & \text{if} \quad f_k(a_j) - f_k(a_i) \leq q(f_k) \\ \dfrac{p(f_k) + f_k(a_i) - f_k(a_j)}{p(f_k) - q(f_k)} & \text{otherwise} \end{cases} \tag{4.2}$$

Algorithm 2: ELECTRE III Algorithm

```
input: n (alternatives), q (criteria), f_i(a_j) (evaluation table);
for all pairs of alternatives a_i, a_j, i,j ∈ {1,...,n} do
    for all criteria f_k, k ∈ {1,...,q} do
        compute outranking degree C_k(a_i, a_j)
    end
end
for all pairs of alternatives a_i, a_j, i,j ∈ {1,...,n} do
    compute concordance index C(a_i, a_j)
end
for all pairs of alternatives a_i, a_j, i,j ∈ {1,...,n} do
    for all criteria f_k, k ∈ {1,...,q} do
        compute disconcordance index D_k(a_i, a_j)
    end
end
for all pairs of alternatives a_i, a_j, i,j ∈ {1,...,n} do
    compute degree of outranking relationship S(a_i, a_j)
end
for all alternatives a_i, i ∈ {1,...,n} do
    compute concordance credibility index φ+(a_i)
    compute disconcordance credibility index φ-(a_i)
    compute net credibility index φ(a_i)
end
FinalRanking = SortDesc(φ)
Result: FinalRanking
```

Step 2: Concordance Index The concordance index $C(a_i, a_j)$ is computed for each pair of alternatives a_i, a_j, as follows:

$$C(a_i, a_j) = \frac{\sum_{k=1}^{q} w_k C_k(a_i, a_j)}{\sum_{k=1}^{q} w_k} \tag{4.3}$$

Step 3: Disconcordance Index Let $v(f_k)$ represent the veto threshold for criterion f_k. The veto threshold rejects the possibility of $a_i S a_j$ if, for any criterion f_k, the relationship $f_k(a_j) > f_k(a_i) + v(f_k)$ is satisfied. The discordance index $D(a_i, a_j)$, $(0 \leq D_k(a_i, a_j) \leq 1)$ for each criterion is defined as follows:

$$D_k(a_i, a_j) = \begin{cases} 0 & \text{if} \quad f_k(a_j) - f_k(a_i) \leq p(f_k) \\ 1 & \text{if} \quad f_k(a_j) - f_k(a_i) > q(f_k) \\ \dfrac{f_k(a_j) - f_k(a_i) - p(f_k)}{v(f_k) - p(f_k)} & \text{otherwise} \end{cases} \tag{4.4}$$

Step 4: Degree of Outranking Relationship Let $J(a_i, a_j)$ represent the set of criteria for which $D_k(a_i, a_j) > C(a_i, a_j)$. The degree of outranking $S(a_i, a_j)$ is:

$$S(a_i, a_j) = \begin{cases} C(a_i, a_j) & \text{if} \quad D_k(a_i, a_j) \leq C(a_i, a_j) \ \forall k \in J \\ C(a_i, a_j) \times \prod_{k \in J(a_i, a_j)} \dfrac{1 - D_k(a_i, a_j)}{1 - C(a_i, a_j)} & \text{otherwise} \end{cases} \tag{4.5}$$

Step 5: Concordance and Disconcordance Credibility Degrees The concordance credibility degree $\phi^+(a_i)$ is an indicator that measures how an alternative a_i dominates all the other alternatives. The definition of concordance credibility degree is:

$$\phi^+(a_i) = \sum_{x \in A} S(a_i, x) \tag{4.6}$$

The disconcordance credibility degree $\phi^-(a_i)$ is an indicator that measures how an alternative a_i is dominated by all the other alternatives. The definition of disconcordance credibility degree is:

$$\phi^-(a_i) = \sum_{x \in A} S(x, a_i) \tag{4.7}$$

Step 6: Net Credibility Degree Finally, the net credibility degree $\phi(a_i)$ is an indicator of the value of the alternative a_i. A higher net credibility degree implies a better alternative. The definition of the net credibility degree for an alternative a_i is:

$$\phi(a_i) = \phi^+(a_i) - \phi^-(a_i) \tag{4.8}$$

The ELECTRE III final ranking is obtained by ordering the alternatives according to the decreasing values of the net flow scores.

4.3.2 PROMETHEE II

The PROMETHEE I and PROMETHEE II methods were introduced by J.P. Brans in 1982 at a conference at the Université Laval, Québec, Canada (L'Ingéniérie de la Décision. Elaboration d'instruments d'Aide à la Décision) and have been extensively applied since then in fields such as business, healthcare and education. The acronym PROMETHEE stands for Preference Ranking Organization METHod for Enrichment of Evaluations. PROMETHEE I is a partial ranking of the actions, as it is based on the positive and negative flows. It includes preferences, indifferences and incomparabilities. On the contrary, PROMETHEE II is a complete ranking of the actions, as it is based on the multicriteria net flow. It includes a preference threshold and an indifference threshold which will be explained in the following paragraphs.

The algorithm of the PROMETHEE II method is presented below:

Algorithm 3: PROMETHEE II Algorithm

input: n (alternatives), q (criteria), $f_i(a_j)$ (evaluation table), P_i (preference funcion);
for *all criteria* f_k, $k \in \{1, \ldots, q\}$ **do**
  $d_k(a_i, a_j) = f_k(a_i) - f_k(a_j)$
end
for *all pairs of alternatives* a_i, a_j, $i, j \in \{1, \ldots, n\}$ **do**
  $\pi_k(a_i, a_j) = P_k[d_k(a_i, a_j)]$
end
for *all criteria* f_k, $k \in \{1, \ldots, q\}$ **do**
  $\pi(a_i, a_j) = \mathbf{sum}(\pi_k(a_i, a_j))$
end
for *all pairs of alternatives* a_i, a_j, $i, j \in \{1, \ldots, n\}$ **do**
  $\phi^+(a_i) = \mathbf{sum}(\pi(a_i, a_j))$
  $\phi^-(a_i) = \mathbf{sum}(\pi(a_j, a_i))$
end
for *all alternatives* a_i, $i \in \{1, \ldots, n\}$ **do**
  $\phi(a_i) = \phi^+(a_i) - \phi^-(a_i)$
end
$FinalRanking = \mathbf{SortDesc}(\phi)$
Result: FinalRanking

Step 1: Pairwise comparisons Firstly, pairwise comparisons are made between all the alternatives for each criterion. $d_k(a_i, a_j)$ is the difference between the evaluations of alternatives a_i and a_j for criterion f_k:

$$d_k(a_i, a_j) = f_k(a_i) - f_k(a_j) \tag{4.9}$$

Step 2: Preference degree The differences calculated in step 1 are translated to preference degrees, according to the selected criterion, as follows:

$$\pi_k(a_i, a_j) = P_k[d_k(a_i, a_j)] \tag{4.10}$$

where $P_k : R \to [0, 1]$ is a positive non-decreasing preference function such that $P_j(0) = 0$. Six different types of preference functions are proposed by PROMETHEE method. These functions are presented at the end of the section.

Step 3: Multicriteria preference degree The pairwise comparison of the alternatives is completed computing the multicriteria preference degree of each pair, as follows:

$$\pi(a_i, a_j) = \sum_{i=1}^{q} \pi_k(a_i, a_j) \cdot w_k \tag{4.11}$$

where w_k represents the weight of criterion f_k, making the assumption that $w_k \geq 0$ and $\sum_{k=1}^{q} w_k = 1$.

Step 4: Multicriteria preference flows Let us define the two following outranking flows:
The positive outranking flow expresses how an alternative is outranking all the others, demonstrating its outranking character. A higher positive outranking flow implies a better alternative.

$$\phi^{+}(a) = \frac{1}{n-1} \sum_{x \in A} \pi(a, x) \tag{4.12}$$

The negative outranking flow expresses how an alternative is outranked by all the others, demonstrating its outranked character. A lower positive outranking flow implies a better alternative.

$$\phi^{-}(a) = \frac{1}{n-1} \sum_{x \in A} \pi(x, a) \tag{4.13}$$

An ideal alternative would have a positive outranking flow equal to 1 and a negative outranking flow equal to 0. The positive and negative outranking flows are aggregated into the net preference flow:

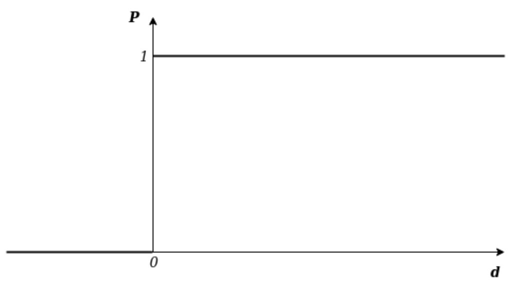
$$\phi(a) = \phi^{+}(a) - \phi^{-}(a) \tag{4.14}$$

The PROMETHEE II final ranking is obtained by ordering the alternatives according to the decreasing values of the net flow scores.

Promethee preference functions Let d_j be the difference of two alternatives a_i, a_j and let q_j, p_j be the indifference and preference thresholds. These parameters are exlained, as follows: when the difference d_j is smaller than the indifference threshold, then it is considered as negligible, therefore the preference degree becomes equal to zero. On the contrary, if the difference is larger than the preference threshold, then it is considered to be significant, therefore the preference degree is equal to one. Otherwise, if the difference is between the two thresholds, the preference degree is computed using a linear interpolation. The six criteria are presented below:

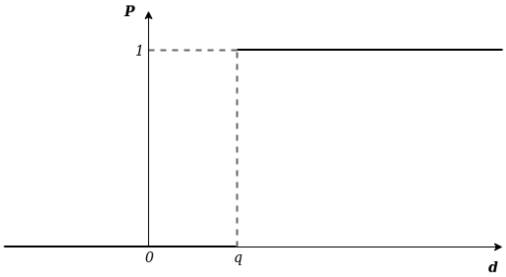
Usual Criterion

$$P_j(d_j) = \begin{cases} 0 & \text{if } d_j \leq 0 \\ 1 & \text{if } d_j > 0 \end{cases}$$



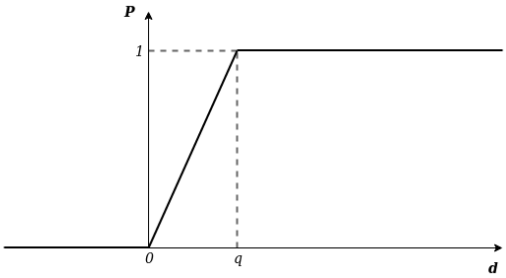
U-shape Criterion

$$P_j(d_j) = \begin{cases} 0 & \text{if} \quad |d_j| \leq q_j \\ 1 & \text{if} \quad |d_j| > q_j \end{cases}$$



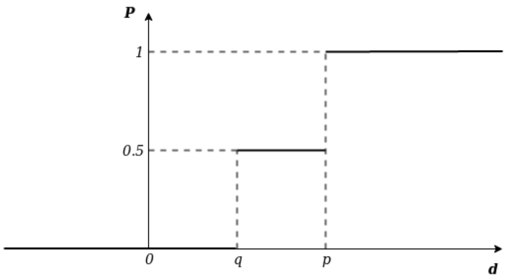
V-shape Criterion

$$P_j(d_j) = \begin{cases} \dfrac{|d_j|}{p_j} & \text{if} \quad |d_j| \leq p_j \\ 1 & \text{if} \quad |d_j| > p_j \end{cases}$$



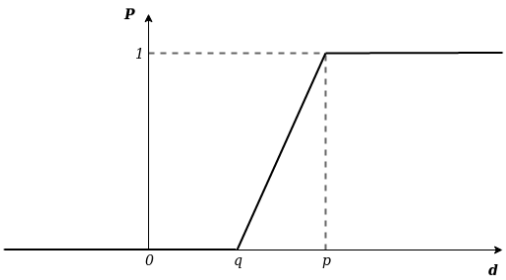
Level Criterion

$$P_j(d_j) = \begin{cases} 0 & \text{if} \quad |d_j| \leq q_j \\ \frac{1}{2} & \text{if} \quad q_j < |d_j| \leq p_j \\ 1 & \text{if} \quad |d_j| > p_j \end{cases}$$



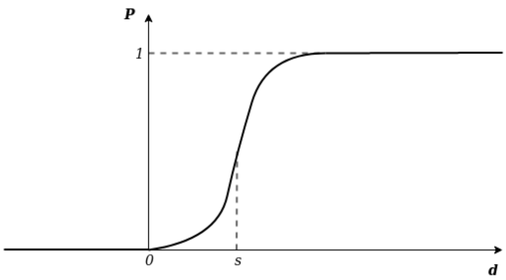
Linear Criterion

$$P_j(d_j) = \begin{cases} 0 & \text{if} \quad |d_j| \leq q_j \\ \dfrac{|d_j| - q_j}{p_j - q_j} & \text{if} \quad q_j < |d_j| \leq p_j \\ 1 & \text{if} \quad |d_j| > p_j \end{cases}$$



Gaussian Criterion

$$P_j(d_j) = 1 - e^{-\dfrac{d_j^2}{2s_j^2}}$$



4.3.3 MAUT

Multi-Attribute Utility Theory (MAUT) (Keeney and Raiffa, 1993) [10] is a structured methodology which was originally designed in order to handle the tradeoffs among multiple objective functions. It was originally developed by Keeney and Raiffa in 1993. MAUT belongs to the family of multiple-criteria utility theory and it has the advantage that it is adaptable to the profile of the DM, as it can describe optimistic and pessimistic behaviours.

The algorithm of the MAUT method is presented below:

Algorithm 4: MAUT Algorithm

input: n (alternatives), q (criteria), w_i (weights), $f_i(a_j)$ (evaluation table);
for *all alternatives* a_i, $i \in \{1, \ldots, n\}$ **do**
  for *all criteria* f_k, $k \in \{1, \ldots, q\}$ **do**
    compute normalised decision matrix $x_k(a_i)$
  end
end
for *all alternatives* a_i, $i \in \{1, \ldots, n\}$ **do**
  for *all criteria* f_k, $k \in \{1, \ldots, q\}$ **do**
    compute integrated DM's attitude $u_k(a_i)$
  end
end
for *all alternatives* a_i, $i \in \{1, \ldots, n\}$ **do**
  compute utility $U(i) = \textbf{sum}(w_k u_k(a_i))$
end
$FinalRanking = \textbf{SortDesc}(U)$
Result: FinalRanking

Step 1: Normalised Decision Matrix Let $f_k(a_{min})$, $f_k(a_{min})$ represent the minimum and maximum value for criterion k. The evaluation table is normalised, as follows:
For maximisation criteria:

$$x_k(a_i) = \frac{f_k(a_i) - f_k(a_{min})}{f_k(a_{max}) - f_k(a_{min})} \tag{4.15}$$

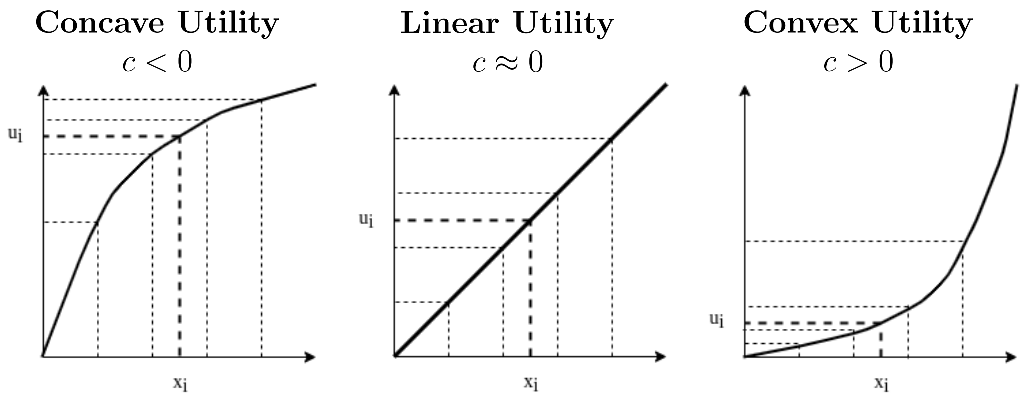
For minimisation criteria:

$$x_k(a_i) = \frac{f_k(a_{max}) - f_k(a_i)}{f_k(a_{max}) - f_k(a_{min})} \tag{4.16}$$

Step 2: Integration of DM Attitude The attitude of the decision-maker is incorporated into the normalised decision matrix, as follows:

$$u_k(a_i) = \frac{1 - e^{cx_i}}{1 - e^c}$$

where c is an index that represents the attitude of the decision maker.



Step 3: Utility Function The Utility function is computed as follows:

$$U_i = \sum_{k=1}^{q} w_k u_k(a_i) \tag{4.17}$$

The MAUT final ranking is obtained by ordering the alternatives according to the decreasing values of the utility function.

4.3.4 TOPSIS

TOPSIS (Hwang and Yoon, 1981) [16] is a multi-criteria decision analysis method. It was originally developed by Ching-Lai Hwang and Yoon in 1981 and further developped by Yoon in 1987, and Hwang, Lai and Liu in 1993. The acronym TOPSIS stands for Technique for Order of Preference by Similarity to Ideal Solution. It is based on the geometric distance from the positive ideal solution (PIS) and the negative ideal solution (NIS). A good alternative has a short distance from the PIS and a long distance from the NIS.

The algorithm of the TOPSIS method is presented below:

Step 1: Normalised Decision Matrix Given the $n \times q$ decision matrix, using the following normalisation method, a new normalised decision matrix is calculated:

$$r_k(a_i) = \frac{x_k(a_i)}{\sqrt{\sum_{j=1}^{n} x_k(a_j)^2}} \tag{4.18}$$

Step 2: Weighted Normalised Decision Matrix In this step, the offsets are incorporated in order to quantify the relative importance of the different criteria. The weighted decision matrix is constructed by multiplying each element of each column of the normalized decision matrix by the offsets:

$$t_k(a_i) = r_k(a_i) \cdot w_k, \quad \text{where } w_k = W_k \Big/ \sum_{j=1}^{q} W_j, \; j = 1, 2, \ldots, q \tag{4.19}$$

Algorithm 5: TOPSIS Algorithm

```
input: n (alternatives), q (criteria), w_i (weights), f_i(a_j) (evaluation table);
for all alternatives a_i, i ∈ {1, ..., n} do
    for all criteria f_k, k ∈ {1, ..., q} do
        compute normalised score r_k(a_i)
    end
end
for all alternatives a_i, i ∈ {1, ..., n} do
    for all criteria f_k, k ∈ {1, ..., q} do
        compute weighted normalised score t_k(a_i)
    end
end
compute positive ideal solution A+
compute negative ideal solution A−
for all alternatives a_i, i ∈ {1, ..., n} do
    compute separation distance from positive ideal solution S+(i)
    compute separation distance from negative ideal solution S−(i)
    compute relative closeness to the positive ideal solution C−(i)
end
FinalRanking = SortDesc(C)
Result: FinalRanking
```

Step 3: Positive and Negative Ideal Solution The positive ideal A^+ and the negative ideal A^- solutions are defined according to the weighted normalised decision matrix:

$$A^+ = \{\langle \min(t_k(a_i) \mid i = 1, 2, \ldots, n) \mid k \in J_- \rangle, \langle \max(t_k(a_i) \mid i = 1, 2, \ldots, n) \mid k \in J_+ \rangle\} \tag{4.20}$$

$$A^- = \{\langle \max(t_k(a_i) \mid i = 1, 2, \ldots, n) \mid k \in J_- \rangle, \langle \min(t_k(a_i) \mid i = 1, 2, \ldots, n) \mid k \in J_+ \rangle\} \tag{4.21}$$

where,

$$J_+ = \{k = 1, 2, \ldots, q \mid k\}$$

for maximisation criteria and

$$J_- = \{k = 1, 2, \ldots, q \mid k\}$$

for minimisation criteria.

Step 4: Separation Distance from the Ideal and Non-Ideal Solution Let t_k^+ be the positive ideal value and t_k^- be the positive ideal value for criterion k. The separation distance ($L^2 - Distance$) of each alternative from the ideal and non-ideal solution is calculated:

$$S_i^+ = \sqrt{\sum_{k=1}^{q} (t_k(a_i) - t_k^+)^2}, \quad i = 1, 2, \ldots, n, \tag{4.22}$$

$$S_i^- = \sqrt{\sum_{k=1}^{q} (t_k(a_i) - t_k^-)^2}, \quad i = 1, 2, \ldots, n \tag{4.23}$$

Step 5: Relative Closeness to the Ideal Solution Finally, for each alternative the relative closeness to the ideal solution is computed. The TOPSIS final ranking is obtained by ordering the alternatives according to the decreasing values of the relative closeness scores.

$$C_i = S_i^- / (S_i^+ + S_i^-), \quad 0 \leq C_i \leq 1, \quad i = 1, 2, \ldots, n \tag{4.24}$$

4.3.5 Cumulative Ranking

After the application of the four MCDA methods, four ranking lists of the alternatives have been formulated. However, the decision-maker should be provided with a final ranking in order to select the k-best securities among them. The suggested methodology to combine the four rankings is the weighted average measure. More specifically, each ranking method is provided with a weighting factor w_k, $k = \{1, 2, 3, 4\}$. The cummulative ranking index CR_i for alternative i is calculated as follows:

$$CR_i = \sum_{k=1}^{4} w_k r_k \tag{4.25}$$

where r_k represents the ranking of alternative i in method k.

4.4 Phase II: Multiobjective Portfolio Optimisation

In the second phase let us introduce the concept of portfolio optimisation. This section cures the problem of capital allocation to the selected securities. Portfolio optimisation is the process of determining the best combination of the weighting factors of securities with the goal to minimise risk and maximise the profit.

The methodological framework of the first phase is depicted in diagram 4.4

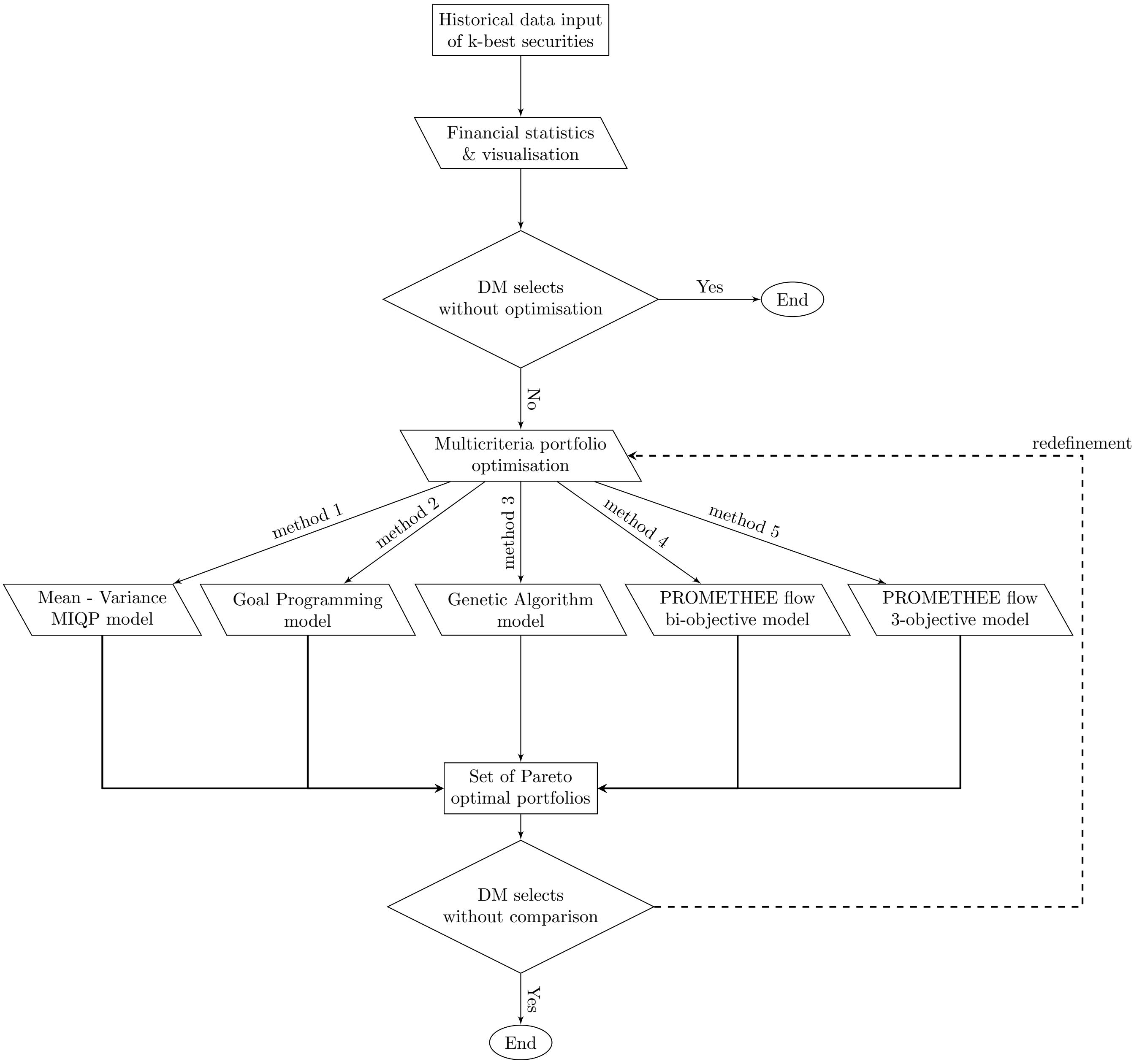


Figure 4.4: Methodological framework for Phase II

The developed methodologies that will be discussed and compared in the following paragraphs are:

1. implementation of the conventional mean-variance model, as well as a mixed-integer variation of the mean-variance method.

2. implementation of a goal programming optimisation model.

3. implementation of a genetic algorithm based on historical data.

4. implementation of bi-objective optimisation problem which involves the PROMETHEE net flow, as well as a MOIP variation of this problem

5. implementation of a 3-objective optimisation problem which involves the PROMETHEE net flow combined with two additional objective functions.

4.4.1 Mean - Variance MIQP Model

The conventional formulation of the portfolio optimisation problem was initially expressed as a nonlinear bi-criteria optimisation problem in 1952. According to Markowitz the portfolio expected return should be maximized and the portfolio risk should be minimized. The risk is quantified as the variance of portfolio returns, resulting in a quadratic programming problem.

Let $E(R_i)$ be the expected return and w_i the weighting factor of security i. The first objective concerns the portfolio expected return and is expressed as follows:

$$\max_{w} E(R_p) = \sum_{i=1}^{m} w_i E(R_i)$$

where m is the total number of securities. Let σ_{ij} be the covariance between securities i and j. The second objective concerns the portfolio risk which is expressed as follows:

$$\min_{w} \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j \neq i}^{m} w_i w_j \sigma_{ij}$$

Moving to the model's set of constraints, the corresponding expression for capital completeness is introduced:

$$\sum_{i=1}^{m} w_i = 1$$

while the restriction concerning no short sales allowance is:

$$w_i \geq 0$$

The above equations constitute a bi-objective quadratic optimisation problem which is presented below:

$$\begin{aligned}
\max_{w} \quad & E(r_P) = \sum_{i=1}^{m} w_i E(r_i) \\
\min_{w} \quad & \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} w_i w_j \sigma_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} w_i = 1 \\
& w_i \geq 0, \qquad i = 1, 2, \ldots, m
\end{aligned} \tag{4.26}$$

The problem is solved parametrically for a predefined parameter of the portfolio expected return. Let R be the portfolio expected return. The problem is transformed into a linear programming problem with an additional restriction concerning the expected return, which is presented below:

$$\begin{aligned}
\min_{w} \quad & \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} w_i w_j \sigma_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} w_i E(r_i) = R \\
& \sum_{i=1}^{m} w_i = 1 \\
& w_i \geq 0, \qquad i = 1, 2, \ldots, m
\end{aligned} \tag{4.27}$$

In this section, a variation of the conventional mean-variance model is developed (Xidonas & Mavrotas, 2012) [22]. The model is equipped with binary variables b_i, in order to control the existence of each security in the portfolio. More specifically, if $b_i = 1$ the i_{th} security participates in the portfolio, else if $b_i = 0$ it does not. The use of binary variables allows the direct determination of the number of securities in the portfolio, producing the following cardinality constraint equation:

$$S_L \leq \sum_{i=1}^{m} b_i \leq S_U$$

where S_L and S_U are the minimum and maximum number of securities allowed to participate in the portfolio.

Moreover, the diversification of the portfolio can be supported constraining the upper bound of each security weight. In order to determine the lower and upper weighting factor of each security the following restrictions are introduced:

$$w_i - W_L \times b_i \geq 0, \quad i = 1, 2, ..., m$$

$$w_i - W_U \times b_i \leq 0, \quad i = 1, 2, ..., m$$

where W_L and W_U are the minimum and maximum security weights that are allowed in the portfolio.

Thus, the following multiobjective integer programming (MOIP) problem is formulated:

$$\begin{aligned}
\max_{w} \quad & E(r_P) = \sum_{i=1}^{m} w_i E(r_i) \\
\min_{w} \quad & \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} w_i w_j \sigma_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} b_i \leq S_U \\
& \sum_{i=1}^{m} b_i \geq S_L \\
& \sum_{i=1}^{m} w_i = 1 \\
& w_i - W_L \times b_i \geq 0 \qquad i = 1, 2, ..., m \\
& w_i - W_U \times b_i \leq 0 \qquad i = 1, 2, ..., m
\end{aligned} \tag{4.28}$$

Similarly to the previous problem, the solution is determined parametrically for a predefined parameter of the portfolio expected return. The problem is transformed into a mixed-integer quadratic programming (MIQP) problem with an additional restriction for the expected return, which is presented below:

$$\begin{aligned}
\min_{w} \quad & \sigma_P^2 = \sum_{i=1}^{m} \sum_{j=1, j\neq i}^{m} w_i w_j \sigma_{ij} \\
\text{s.t.} \quad & \sum_{i=1}^{m} w_i E(r_i) = R \\
& \sum_{i=1}^{m} b_i \leq S_U \\
& \sum_{i=1}^{m} b_i \geq S_L \\
& \sum_{i=1}^{m} w_i = 1 \\
& w_i - W_L \times b_i \geq 0 \qquad i = 1, 2, ..., m \\
& w_i - W_U \times b_i \leq 0 \qquad i = 1, 2, ..., m
\end{aligned} \tag{4.29}$$

The formulated problem can be solved parametrically considering the parameter R, thus producing the efficient frontier of solutions.

4.4.2 Goal Programming Model

Another approach to the portfolio optimisation problem is the development of a goal programming model. As discussed in the previous chapter, goal programming is a multiobjective optimisation method which is used to handle problem of conflicting objective functions.

The decision variables of the goal programming problem will be the weighting factor w of each security. Let w_i be the weighting factor of the i_{th} security. The following goals are defined:

1. The beta index of the portfolio β_P, which is defined as the weighted sum of the individual beta index of each security, is given the target value β_G.

$$\beta_P = \sum_{i=1}^{m} w_i \times \beta_i$$

2. The portfolio dividend yield, which is defined as the weighted sum of the individual dividend yield of each security, is given the target value DY_G

$$DY_P = \sum_{i=1}^{m} w_i \times DY_i$$

3. The portfolio PROMETHEE flow, which is defined as the weighted sum of the individual flow of each security, is given the target value ϕ_G

$$\phi_P = \sum_{i=1}^{m} w_i \times \phi_i$$

The model is equipped with binary variables b, in order to control the existence of each security in the portfolio, producing the following cardinality constraint equation:

$$S_L \leq \sum_{i=1}^{m} b_i \leq S_U$$

where S_L and S_U are the minimum and maximum number of securities allowed to participate in the portfolio.

In order to determine the lower and upper weighting factor of each security the following restrictions are introduced:

$$w_i - W_L \times b_i \geq 0, \quad i = 1, 2, ..., m$$

$$w_i - W_U \times b_i \leq 0, \quad i = 1, 2, ..., m$$

where W_L and W_U are the minimum and maximum security weights that are allowed in the portfolio.

Introducing the deviational (or slack) variables d_i^+, d_i^- the problem is formulated as follows:

$$
\begin{aligned}
\min_{d^+,d^-} \quad & \frac{w_1^+ d_1^+ + w_1^- d_1^-}{\beta_G} + \frac{w_2^+ d_2^+ + w_2^- d_2^-}{DY_G} + \frac{w_3^+ d_3^+ + w_3^- d_3^-}{\phi_G} \\
\text{s.t.} \quad \sum_{i=1}^{m} w_i \beta_i + d_1^- - d_1^+ &= \beta_G \\
\sum_{i=1}^{m} w_i DY_i + d_2^- - d_2^+ &= DY_G \\
\sum_{i=1}^{m} w_i \phi_i + d_3^- - d_3^+ &= \phi_G \\
\sum_{i=1}^{m} b_i &\leq S_U \\
\sum_{i=1}^{m} b_i &\geq S_L \\
\sum_{i=1}^{m} w_i &= 1 \\
w_i - W_L \times b_i &\geq 0 \qquad i = 1, 2, ..., m \\
w_i - W_U \times b_i &\leq 0 \qquad i = 1, 2, ..., m
\end{aligned}
\tag{4.30}
$$

where w^+, w^- are the weights of the deviational variables. Attention is needed not to confuse the weighting factor w of each security with the overachievement and underachievement weights w^+,w^- of the deviational variables

4.4.3 Genetic Algorithm Model

In this paragraph we introduce an implementation of an alternative model applying a genetic algorithm. The philosophy of this problem differs from all the other, as the weighting factors are determined with the assistance of a market index. Additionally, another significant difference is that in this case there is a unique solution of the optimisation problem, while the other problems result in a set of pareto efficient solutions.

More specifically, let m be the market index and r_{im} the return of index m in period i. Let us define, also, m securities and r_{ij} the return of security j in period i. The portfolio return in period i is equal to:

$$r_{ip} = \sum_{j=1}^{m} w_j r_{ij} \tag{4.31}$$

where w_j is the proportion of the security j in portfolio p.

We say that *the constructed portfolio beats the market index* in period i if the following inequation applies:

$$r_{ip} \geq r_{im} \tag{4.32}$$

Therefore, the genetic algorithm takes as input the historical data for T periods and attemps to maximise the percentage of cases that the constructed portfolio beats the market index. This claim is quantified as follows:

$$\begin{aligned} \max_{w_i} \sum_{i=1}^{T} b_i \Big/ T & \\ \text{s.t.} \sum_{i=1}^{N} w_i &= 1 \\ w_i &\geq 0 \quad \forall i = 1, 2, \ldots N \end{aligned} \tag{4.33}$$

where b_i is a binary variable that takes the value 1 if $r_{ip} \geq r_{im}$ in period i, else it takes the value 0.

Conclusively, the genetic algorithm provides the optimal portfolio proportions, such that the percentage that the constructed portfolio results in better return than the market index is maximised.

4.4.4 MOIP PROMETHEE Flow Model

PROMETHEE Flow Bi-Objective Model

In this paragraph, an alternative approach to the classic mean-variance model is presented. This approach connects the concept of the PROMETHEE method of multiple-criteria decision analysis with a measure of risk, in this case Beta index. The beta index of a portofolio is the average of the beta indexes of the participating securities. Let ϕ_i represent the PROMETHEE net flow of the i_{th} security. The first objective function of the problem involves the maximisation of the average net flow:

$$max_w \quad \phi_P = \sum_{i=1}^{m} w_i \phi_i$$

The second objective of the problem involves the minimisation of portfolio beta index:

$$min_w \quad \beta_P = \sum_{i=1}^{m} w_i \beta_i$$

Therefore, adding to the model the constraints of capital completeness and no short sales allowance, a bi-objective linear programming optimisation problem is formulated as follows:

$$\begin{aligned} \max_{w} \quad & \phi_P = \sum_{i=1}^{m} w_i \phi_i \\ \min_{w} \quad & \beta_P = \sum_{i=1}^{m} w_i \beta_i \\ \text{s.t.} \quad & \sum_{i=1}^{m} w_i = 1 \\ & w_i \geq 0, \quad i = 1, 2, \ldots, m \end{aligned} \tag{4.34}$$

If we introduce binary variables to the above problem, accordingly to the extension of mean - variance model, a bi-objective integer programming problem is formulated as follows:

$$
\begin{aligned}
\max_{w} \quad & \phi_P = \sum_{i=1}^{m} w_i \phi_i \\
\min_{w} \quad & \beta_P = \sum_{i=1}^{m} w_i \beta_i \\
\text{s.t.} \quad & \sum_{i=1}^{m} b_i \leq S_U \\
& \sum_{i=1}^{m} b_i \geq S_L \\
& \sum_{i=1}^{m} w_i = 1 \\
& w_i - W_L \times b_i \geq 0 \qquad i = 1, 2, ..., m \\
& w_i - W_U \times b_i \leq 0 \qquad i = 1, 2, ..., m
\end{aligned}
\tag{4.35}
$$

The above problems can be solved parametrically, transforming one objective function into an additional constraint, accordingly to the conventional mean-variance model.

PROMETHEE Flow 3-Objective Model

Finally, in this paragraph a 3-objective variation of the above model is introduced, in order to incorporate another objective function in the portfolio optimisation problem.

Without loss of generality, let PROMETHEE flow be the first objective function. Alternatively, any other method could be introduced, such as ELECTRE III, MAUT or TOPSIS. We selected the PROMETHEE net flow, because of PROMETHEE's capability to use a variety of functions depending on the criterion. Therefore, the first goal is to optimise the net flow of PROMETHEE as it was introduced in Phase A.

We propose, the second objective to be the Beta index in order to incorporate a measure of risk to the model. Finally, let us introduce the dividend yield as a third objective function.

Additionally, the problem could be equipped with integer decision variables in order to control the number of securities with non-zero proportion to the portfolio.

Based on the above observations the optimisation problem is formulated as follows:

$$\begin{array}{llll}
\max\limits_{w} & \phi_P = \sum\limits_{i=1}^{m} w_i \phi_i & & \\
\min\limits_{w} & \beta_P = \sum\limits_{i=1}^{m} w_i \beta_i & & \\
\max\limits_{w} & DY_P = \sum\limits_{i=1}^{m} w_i DY_i & & \\
\text{s.t.} & \sum\limits_{i=1}^{m} b_i \leq S_U & & \\
& \sum\limits_{i=1}^{m} b_i \geq S_L & & \\
& \sum\limits_{i=1}^{m} w_i = 1 & & \\
& w_i - W_L \times b_i \geq 0 & i = 1, 2, ..., m & \\
& w_i - W_U \times b_i \leq 0 & i = 1, 2, ..., m &
\end{array} \tag{4.36}$$

This is a multiobjective programming problem in 3 dimensions with integer variables. It is obvious that the computational complexity of the above problem becomes huge, especially if the number of securities is significantly large. A variety of methodologies have been proposed for problems like this such as the e-constraint method, which faces the problem as a 1-objective optimisation problem, transforming the remaining objectives into constraints. However, in this paragraph a methodology based on goal programming and the MINIMAX objective is proposed to solve this MOLP problem.

The first step of the methodology is to solve the model to find the solution that minimises each objective function ignoring the other objective function. If we solve the problem for all objective functions, we obtain the optimal value for each objective, respectively.

The next step is to formulate the goal programming problem. The target value for each objective function is set equal to the optimal value calculated in the previous step. The percentage deviation from this target can be computed as follows:

$$t = \frac{actual\ value - target\ value}{target\ value} \tag{4.37}$$

for goals derived from minimisation objectives,

$$t = \frac{target\ value - actual\ value}{target\ value} \tag{4.38}$$

for goals derived from maximisation objectives.

Therefore, having determined the goals of the GP model, the last step involves the configuration of the objective function. The implementation of the objective function is made with the introduction of a MINIMAX variable Q which should be minimised. If w_i is the offset for the i_{th} objective function, the goal is to minimise the maximum of $w_i t_i$. The above claim is expressed with the following mathematical equation:

$$\begin{aligned} &\min Q \\ &\text{s.t.} \quad w_1 t_1 \leq Q \\ &\qquad\;\; w_2 t_2 \leq Q \\ &\qquad\;\; w_3 t_3 \leq Q \end{aligned} \tag{4.39}$$

Thus, a set of pareto optimal solutions derives from the adjustment of the weighting factors w_i.

4.5 Conclusion

In the current chapter, there was a complete presentation of the proposed methodology, which tried to incorporate all the parameters of the problem of efficient portfolio management. The proposed methodology was split in two phases: (a) in the first phase we introduced an integrated methodology for the problem of security selection and (b) in the second phase we presented a variety of alternative methodologies which aim to solving the problem of portfolio optimisation.

For completeness reasons, it is necessary to make a quick discussion about the final decision problematic, concerning the selection of the most suitable portfolio from the set of efficient portfolios. Given a pareto efficient set of candidate portfolios, the problems lies to the determination of the most appropriate one. It is obvious that the most significant parameter of the problem is the profile of the decision-maker. The decision-maker's profile creates a clear picture about how a person makes a decision and determines the way that he should be supported in decision-making. For instance, a risk-averse decision-maker would probably exclude all the portfolios that result in increased portfolio risk, while on the contrary an aggressive decision-maker would select one of the most risky portfolios of the efficient set. Subsequently, the decision-maker might determine the suitable portfolio without any additional support.

However, in case that the investor has not reached to a final decision a methodological framework for decision support of the final phase is developed. The problem of selecting the most dominant portfolio of all the feasible ones can be solved as a discrete MCDA problem, where the alternatives are all the efficient portfolios and the criteria can be set in communication with the decision-maker. Therefore, the methodological framework that was used in Phase I for security selection can be also used to this problem for the determination of the most suitable portfolio. After the application of the MCDA methods a final ranking of the portfolios is obtained and finally the investor has the opportunity to select the most appropriate one.

After the presentation of the suggested methodology for portfolio selection, we can conclude that the proposed methodological framework provides the decision-maker with an integrated decision-support model, considering every aspect of the whole procedure. Additionally, the most significant accomplishment is the incorporation of the investor's preferences and the interaction with the DM at every part of the process.

Chapter 5

Information System

5.1 Introduction

The presentation of the proposed methodology emphasised the need for modern information systems in order to implement the necessary methods. In the context of the current thesis, a decision support information system was designed and developed. The purpose of the information system is to implement efficiently the algorithms described in the previous section in order to support the decision-making process.

The information system consists of four subsystems. The first subsystem includes an implementation of the multiple-criteria decision support methods that are used in the proposed methodology. The second subsystem supports financial statistics calculation. The third subsystem implements the multiobjective programming methods for portfolio optimisation. Finally, the fourth subsystem assists the evaluation process, offering a visualisation of the efficient portfolios.

In addition, the first subsystem is deployed as a web application. The specific platform offers a friendly graphical user interface *GUI* and offers efficient implementations of specific MCDA methods, providing extensive solutions for a wide range of multiple-criteria problems.

5.2 System Architecture and Attributes

In this section, there is a brief introduction to the programming tools which were used, as well as the libraries that supported the visualisation of data and the optimisation methods. The information system is developed in Python 3 programming language, which makes it available for Windows, Linux and macOS operating systems.

Python 3.0 Programming Language

Python is a general-purpose, high level scripting programming language, which is widely used nowadays. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. Its main goals is to provide code readability, and simplify complex concepts. It is an interpreted language, i.e. the steps of code compilation and execution are unified and the program can be directly executed from the source code. Additionally, it is platform independent as it can be used on Linux, Windows, Macintosh and multiple other operating systems. Python can be used for a variety of tasks in many sectors including:

1. Mathematics and physics
2. Quantitative finance and financial econometrics
3. Machine learning, neural networks and artificial intelligence
4. Big data applications and data engineering
5. Network security, prototyping applications
6. Enterprise and business applications, web frameworks and applicaitons etc.

More specifically, Python 3.0 (also called "Python 3000" or "Py3K") was released on December 3, 2008. It has a wide variety of advantages, the most important of which are presented below:

- It incorporates an extensive support of additional libraries, such as NumPy for numerical calculation and Pandas for data analysis.
- It provides object-oriented utilities, it is portable and interactive.
- It is an open source language, with a developed community.
- It is considered an easy to learn programming language, providing user-friendly data structures.

In the following paragraphs we make a quick introduction to the libraries that were used for the implementation of the model. The main libraries which support the development of the information system are matplotlib, pandas, numpy and mip.

Matplotlib [6] is a Python plotting library which produces high quality figures in a variety of formats, such as plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook and web application servers. It was originally written by John D. Hunter, providing an interface with close resemblanse to MatLab and it has an active development community.

Pandas [17] is an open source library which provides high performance, useful data structures and data analysis tools for the Python programming language. Some of the most important included features are an efficient DataFrame object for data manipulation with integrated indexing, tools for reading and writing data between different formats (CSV, text files, Microsoft Excel etc.) and flexible reshaping functionality of data sets. It incorporates significant optimisations providing high performance and it has a wide variety of uses in academic and commercial fields.

NumPy [7] is the fundamental library for scientific computing with Python, as it contains various useful tools. The most important functionalities are the N-dimensional array object, the tools for importing C/C++ and Fortran code, as well a variety of function, such as linear algebra, Fourier transform, and random number capabilities. Additionally, NumPy can also be used as an efficient multi-dimensional container of generic data, allowing NumPy to integrate with a wide variety of databases with very high speed.

MIP is a library of Python tools for the modeling and solution of Mixed-Integer Linear programming problems. Some of the main functionalities provided by MIP are the following: Firstly, high level modeling capability, offering the opportunity of easy implementation of linear relations, as in high level languages. Besides, the Python MIP package is deeply integrated with many solvers, such as Branch-and-Cut and the commercial solver Gurobi.

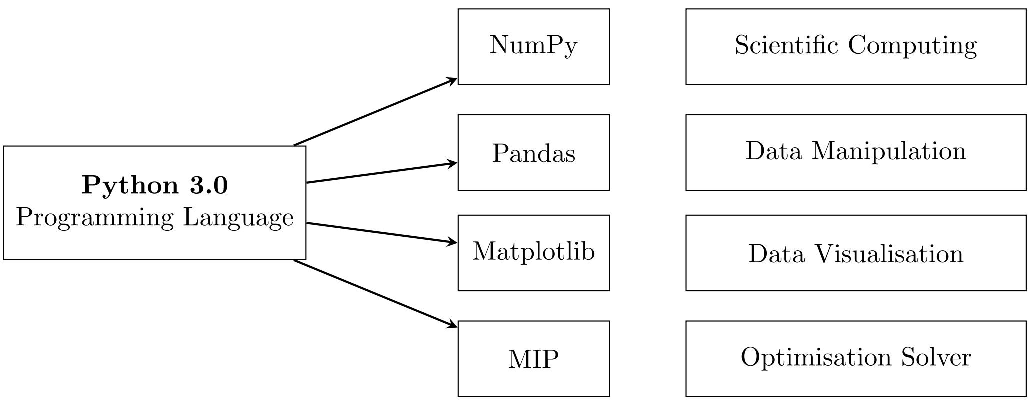


Figure 5.1: Programming language and additional packages

5.3 Interaction Diagrams

In this section, we present the basic UML interaction diagrams of the information system, in order to demonstrate how it can be used. Through these diagrams, the interaction between the decision-maker and the information system is reflected.

Use Case Diagram A use case diagram is a graphic demmonstration of the interactions among the elements of an information system. In the following diagram the three parts of the system are the decision-maker, the information system and the Yahoo Finance database. The use case diagram is presented in figure 5.2:

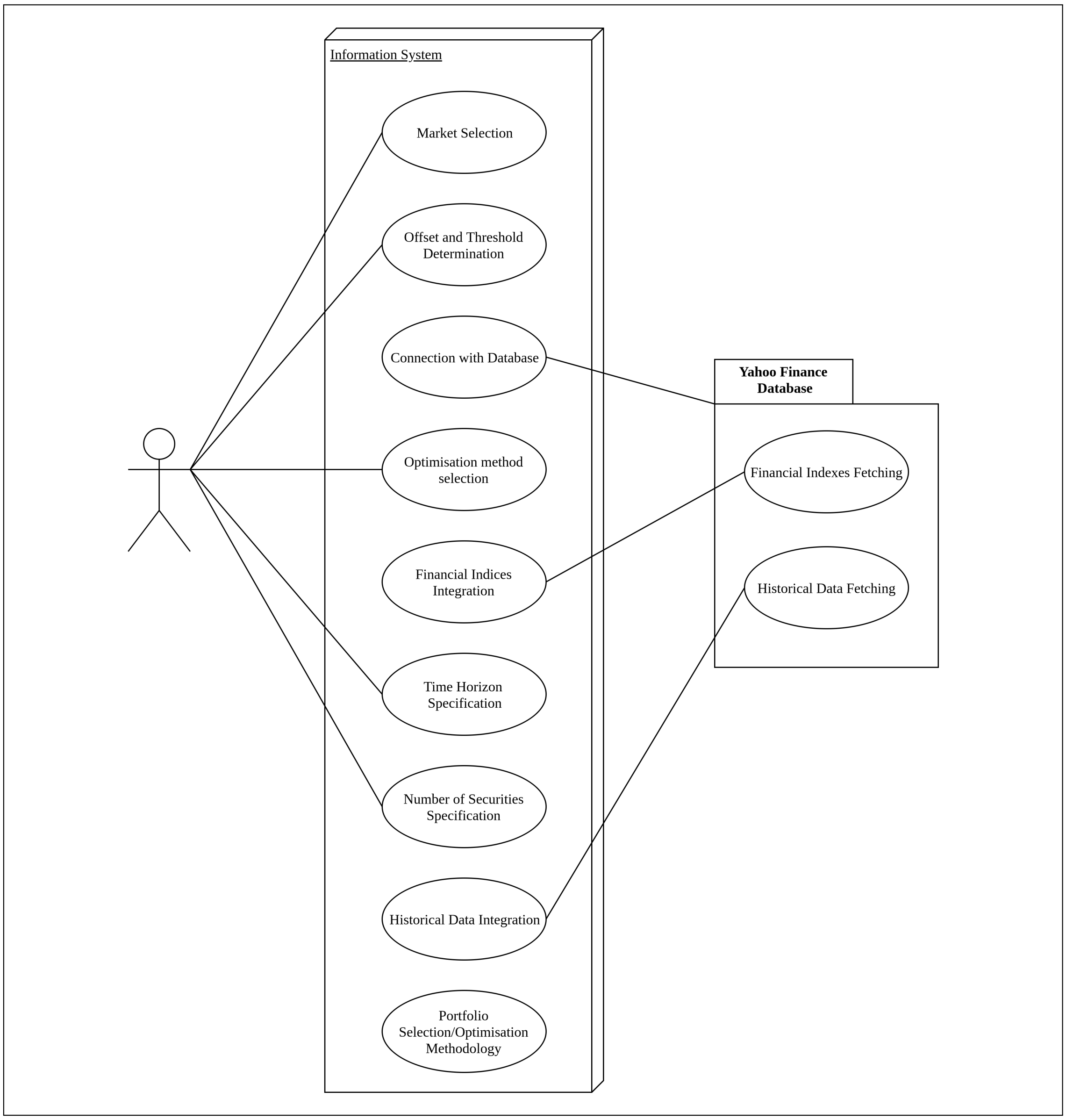


Figure 5.2: UML Use Case Diagram

Sequential Diagram A sequential diagram describes the interactions among each part of the system arranged in time sequence. It shows the sequence of messages which are exchanged between the different parts of the system (decision-maker, database etc.), in order to achieve the functionality of each scenario. The sequential diagram of the system is presented in figure 5.3

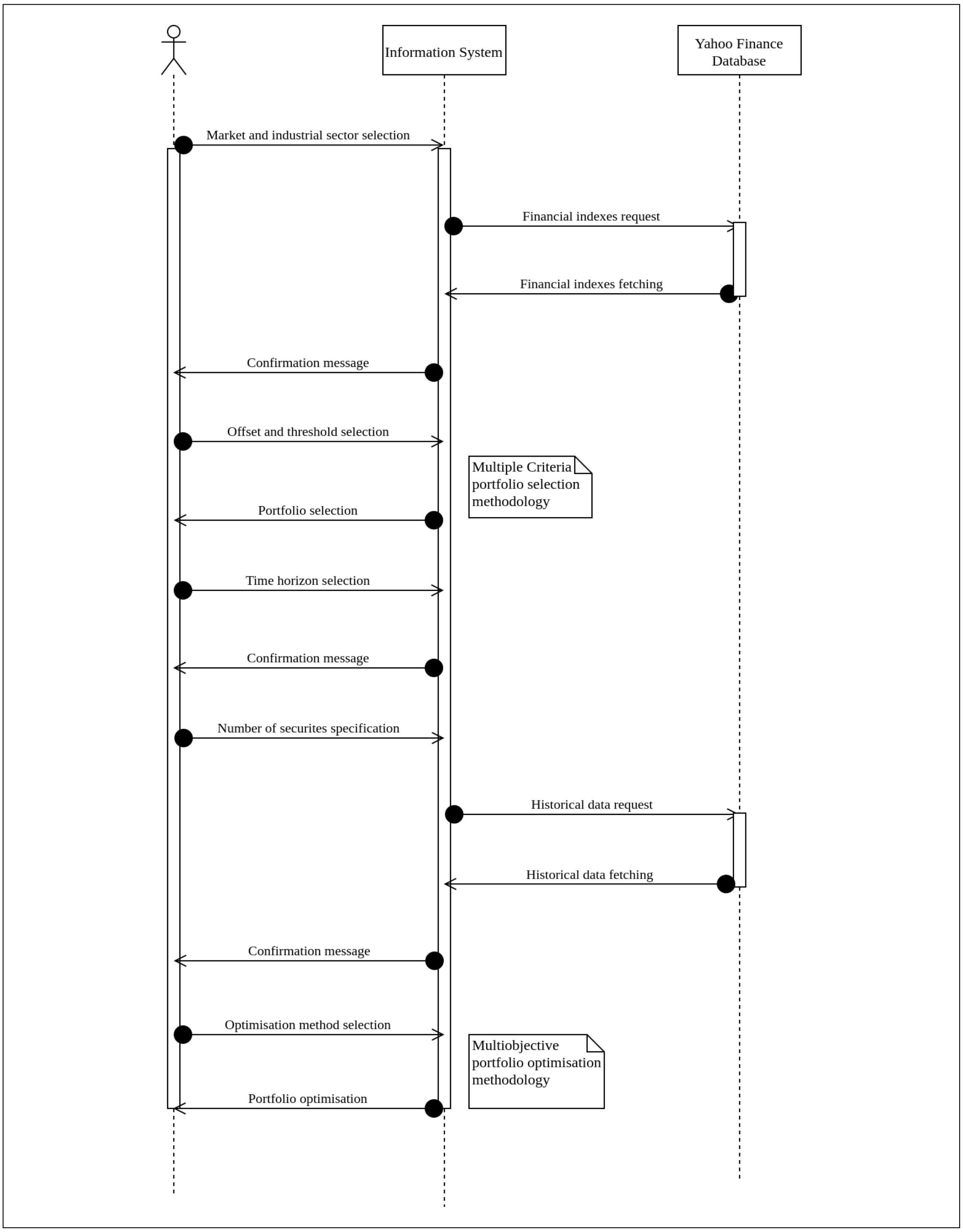


Figure 5.3: UML Sequential Diagram

Communication Diagram A communication diagram describes the interactions between the various parts of the system in the form of sequenced messages. These diagrams are composed of a combination of sequence and use case diagrams incorporating both the static and the dynamic behavior of the information system. The communication diagram of the system is presented in figure 5.4.

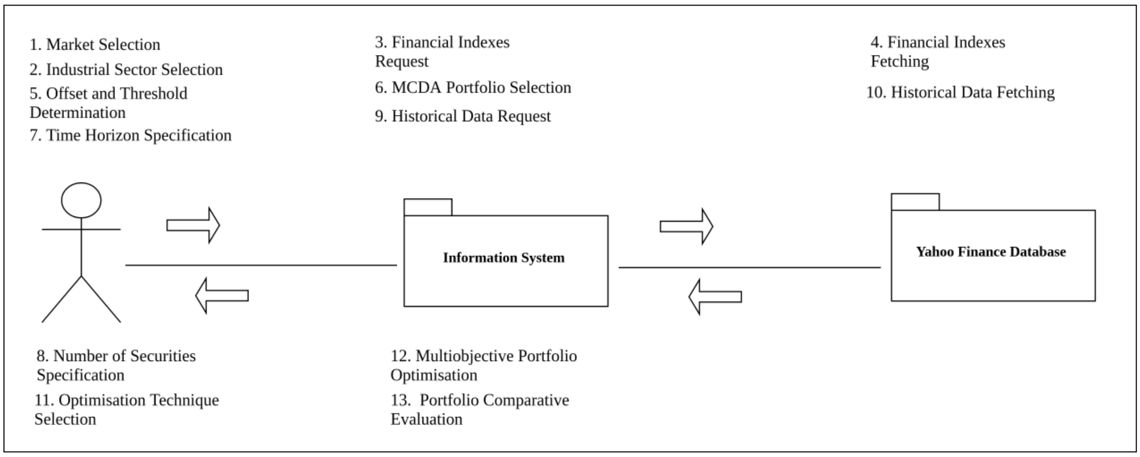


Figure 5.4: UML Communication Diagram

5.4 MCDA Platform

As part of the thesis project, a module of the information system was deployed as a web application. This application includes an efficient implementation of a variety of multicriteria decision analysis methods, such as ELECTRE, PROMETHEE, MAUT and TOPSIS. This project is implemented in one of the most used Python web frameworks called *Django*.

Django is a high-level Python free and open-source web framework which supports rapid development combined with modern design. It follows the conventional model-template-view web arcitecture and it is used for the creation of database-driven websites. Django framework is generally based to an MVC architecture, as it consists of an object-relational mapper (ORM) that mediates between data models. Additionally, it includes a relational database called *Model*, a subsystem for processing HTTP requests known as *View* and a URL dispatcher called *Controller*.

In this paragraph, we introduce some of the most important features of the MCDA platform. Initially, the homepage of the system is designed in order to present the application's content (Figure 5.5).

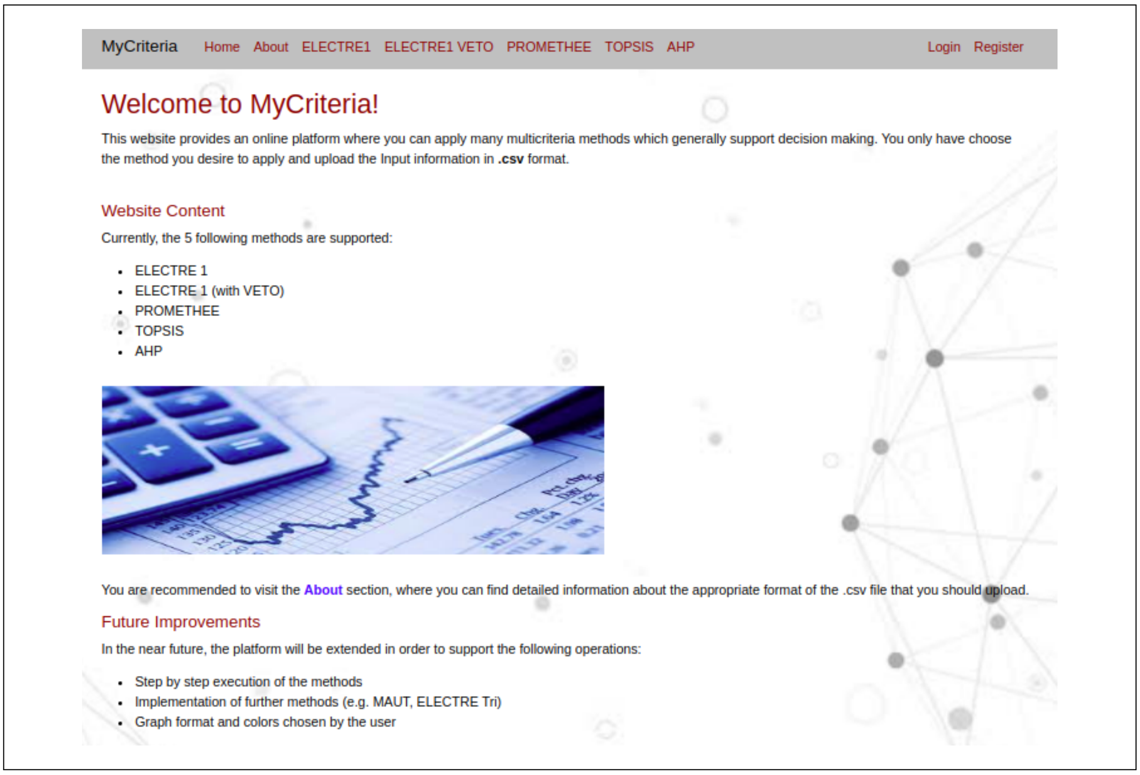


Figure 5.5: Homepage screen

The user is able to select one of the supported MCDA methods and have access to the detailed information and the input prerequisites of the selected method

(Figure 5.6).

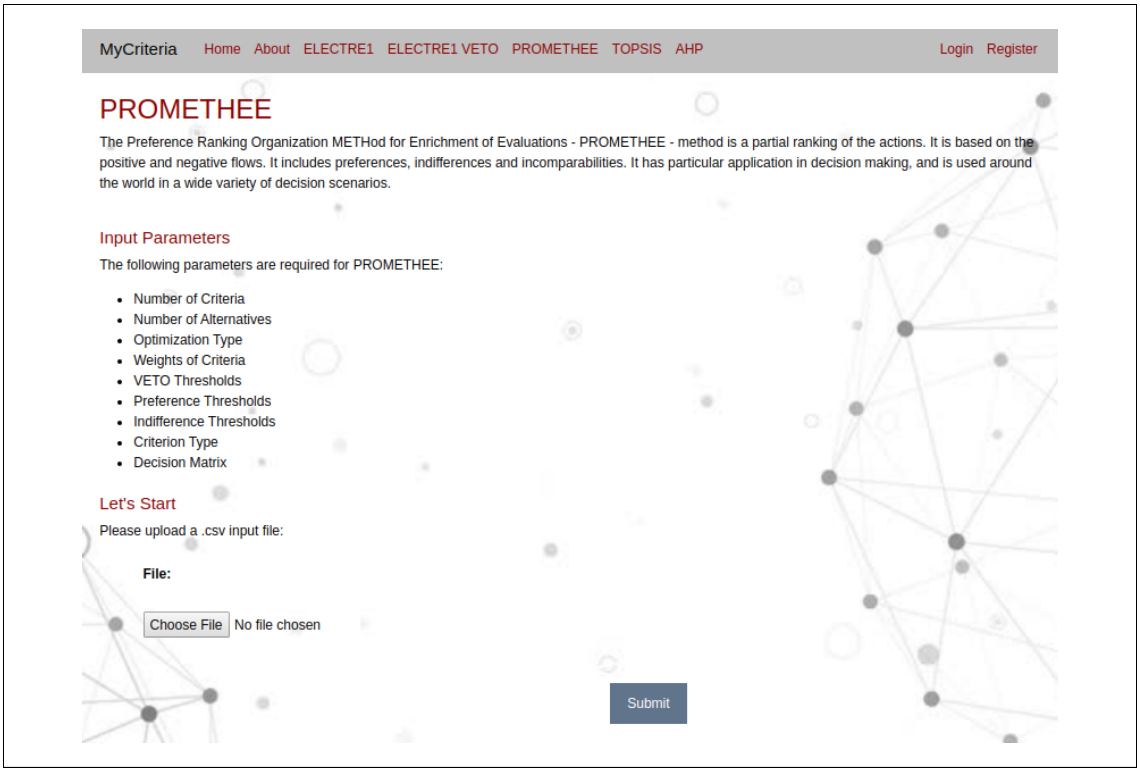


Figure 5.6: Individual method page screen

The platform supports all necessary methods for the proposed methodology, which are (a) ELECTRE III (b) PROMETHEE II (c) MAUT and (d) TOPSIS. Except from these ranking methods, the platform also supports two choice methods: ELECTRE I and ELECTRE VETO. Additionally, there is an *About* page which refers to the format of the necessary input (Figure 5.7).

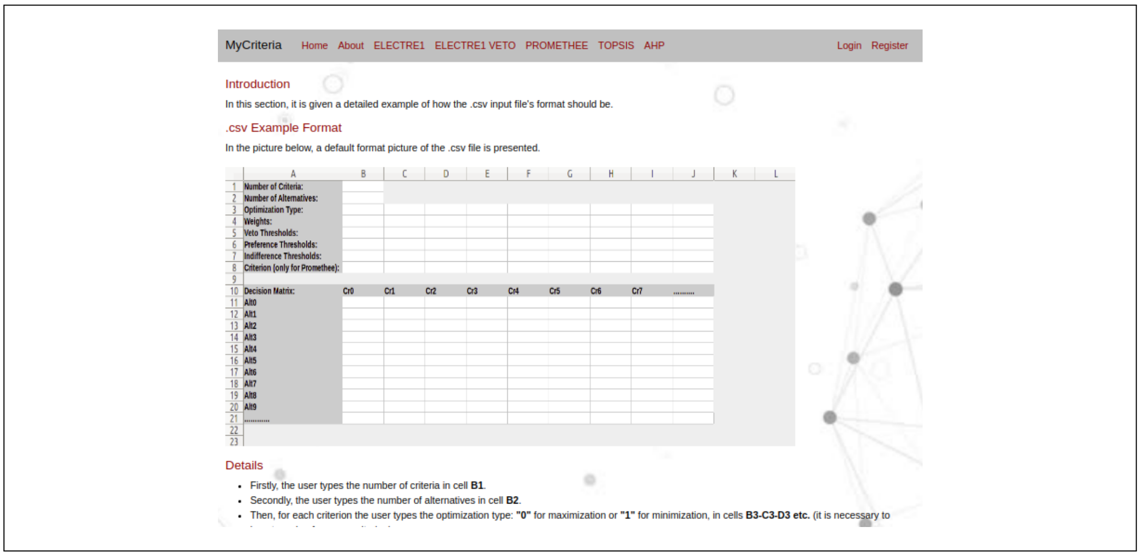
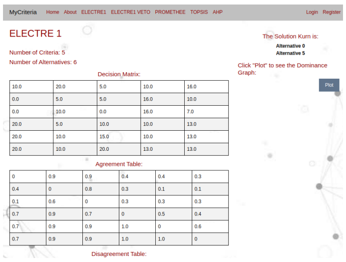
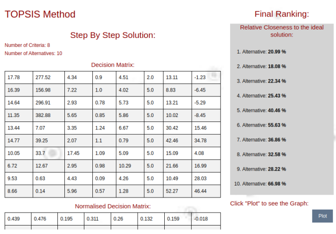


Figure 5.7: About page screen

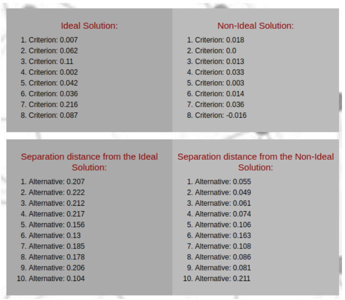
The input can be given in *.csv* or *.xls* format to the platform. Except from the general information (alternatives, criteria, decision matrix and weights), different additional information should be imported according to the method. For example, in PROMETHEE methods, it is necessary the criterion type and the appropriate thresholds to be determined. After the procedure of commiting the input file to te platform, the next stage incorporates the presentation of the results. In the following figures, you can see the output screen for the application of some of the supported methods (Figure 5.8).



(a) ELECTRE I



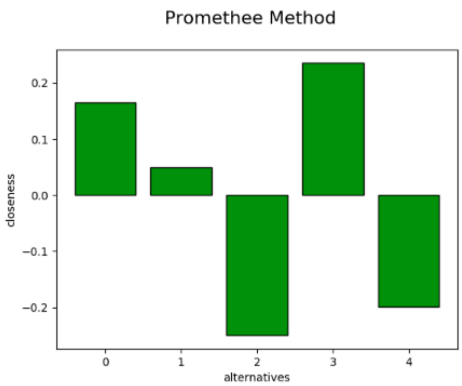
(b) TOPSIS (i)



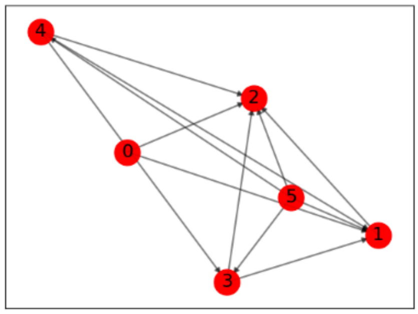
(c) TOPSIS (ii)

Figure 5.8: Results Presentation Screen

Finally a visualisation of the results can be produced by clicking the corresponding button at the results page. The visualisation depends on the method and it is adjusted to the method category. For instance, for a ranking method a barplot with the final ranking of the alternatives is produced, while for a choice method the dominance graph is alternatively produced, showing which alternatives dominate over the others (Figure 5.9).



(a) Ranking method



(b) Choice method

Figure 5.9: Results Visualisation Screen

5.5 Source Code Presentation

In this section, we present and explain a part of the source code of the information system. More specifically, the implementation of portfolio optimisation process is presented. It includes the connection with the Yahoo API, the manipulation of the historical data, the calculation of the most significant statistical indexes and finally the portfolio optimisation procedure based on the mean-variance model.

The whole source code, including the MCDA methods as well as the alternative optimisation techniques is presented in detail in the appendix. The presentation of the code is made using the *Jupiter Notebook* application which is designed to support the interactive development and presentation of data science projects. Therefore, the source code presentation begins in the following paragraph:

[1]:
```
from pandas_datareader import data
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
import scipy.optimize as sco
```

Data Fetching

Determination of parameters

After importing all necessary libraries, we must define the securities, as well as the starting and ending date of the empirical testing. The list tickers contains the names of the securities and the variables startDate, endDate contain the duration of the simulation. As you can see, the selected time horizon is 4 years, from 1-1-2016 until 31-12-2019 and the selected equities of this example are 6 securities from the energy sector of NYSE stock exchange.

[2]:
```
tickers = ['BP', 'CELP', 'CEO', 'CVI', 'CVX', 'CZZ']
startDate = '2016-01-01'
endDate = '2019-12-31'
```

Connection to the Yahoo API

In the following part, we make the connection with the yahoo! finance API. This connection is made with the data.DataReader function of the pandas library. It takes four arguments: (a) a list with the names of the securities, (b) the starting date, (c) the ending date and (d) the name of the API (in this case 'yahoo'). For example, in the following cell you can see the historical values of the first security 'BP'. The result contains the values 'High', 'Low', 'Open', 'Close', 'Volume' and 'Adj Close'.

[3]:
```
historicalDataBP = data.DataReader('BP', 'yahoo', startDate,␣
    ↪endDate)
print("==========  Raw Data from Yahoo API  ========== \n")
display(historicalDataBP)
```

```
====================  Raw Data from Yahoo API  ====================


                 High        Low       Open      Close      Volume  Adj Close
Date
2016-01-04  31.170000  30.510000  30.799999  31.059999   7582300.0  23.981899
2016-01-05  30.990000  30.379999  30.920000  30.930000   7234400.0  23.881523
2016-01-06  30.410000  29.930000  29.930000  30.299999  10055100.0  23.395094
2016-01-07  29.820000  29.000000  29.070000  29.430000  15156000.0  22.723352
2016-01-08  29.450001  28.840000  29.410000  28.910000  13901300.0  22.321848
...               ...        ...        ...        ...         ...        ...
2019-12-24  38.139999  37.959999  37.970001  38.040001   2348400.0  38.040001
2019-12-26  38.200001  37.939999  38.060001  37.980000   4504200.0  37.980000
2019-12-27  38.250000  37.860001  38.240002  37.860001   5436600.0  37.860001
2019-12-30  37.970001  37.570000  37.799999  37.599998   6105900.0  37.599998
2019-12-31  37.480000  37.330101  37.799999  37.439999    432547.0  37.439999

[1006 rows x 6 columns]
```

Data Pre-processing

From all the security values we only need the data of the 'Open' column in order to use it for the empirical testing. Therefore, we firstly draw all the desired data from the Yahoo API and then we discard the unnecessary columns, as follows:

[4]:
```
historicalValues = data.DataReader(tickers, 'yahoo', startDate,␣
    ↪endDate)
stockValues = historicalValues['Open']
numOfDates = stockValues.shape[0]
numOfSecurities = stockValues.shape[1]
```

```
print("Number of securities:", numOfSecurities)
print("Number of dates:", numOfDates, "\n")
print("==============  Stock Values  ============== \n")
display(stockValues)
```

```
Number of securities: 6
Number of dates: 1006

=====================  Stock Values  =====================



Symbols             BP  CELP         CEO        CVI         CVX        CZZ
Date
2016-01-04   30.799999  8.80  103.059998  38.880001   89.529999   3.590000
2016-01-05   30.920000  8.64  103.500000  37.970001   89.050003   3.550000
2016-01-06   29.930000  8.25  101.250000  37.759998   87.440002   3.330000
2016-01-07   29.070000  8.10   97.790001  36.740002   84.550003   3.090000
2016-01-08   29.410000  7.49   97.220001  37.389999   83.389999   3.050000
...                ...   ...         ...        ...         ...        ...
2019-12-24   37.970001  9.60  163.460007  41.320000  120.430000  22.250000
2019-12-26   38.060001  9.65  163.229996  41.009998  120.669998  23.230000
2019-12-27   38.240002  9.40  165.500000  41.049999  120.889999  23.450001
2019-12-30   37.799999  9.33  165.369995  40.369999  120.440002  22.940001
2019-12-31   37.799999  9.33  165.369995  40.369999  120.440002  22.940001

[1006 rows x 6 columns]
```

Security Values Visualisation

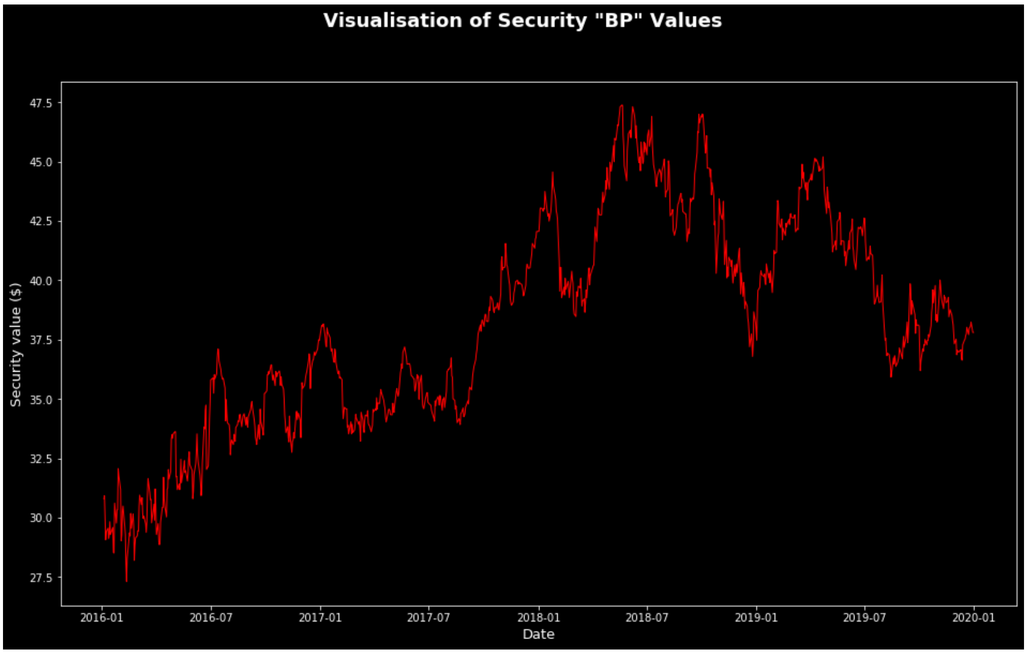
Individual diagram for one security

Now, the dataframe stockValues contains the historical values of the securities. These values can be easily visualised with matplotlib library. The visualisation of security ‘BP’ is presented below:

[5]:
```
from pandas.plotting import register_matplotlib_converters
register_matplotlib_converters()

plt.style.use('dark_background')

fig, ax = plt.subplots(figsize=(16,9))
```

```
plt.suptitle('Visualisation of Security "BP" Values', fontsize=18,␣
 ↪fontweight='bold')
ax.plot(stockValues.index, stockValues['BP'], color='red', lw=1)
ax.set_xlabel('Date', fontsize=13)
ax.set_ylabel('Security value ($)', fontsize=13)
```

[5]: Text(0, 0.5, 'Security value ($)')

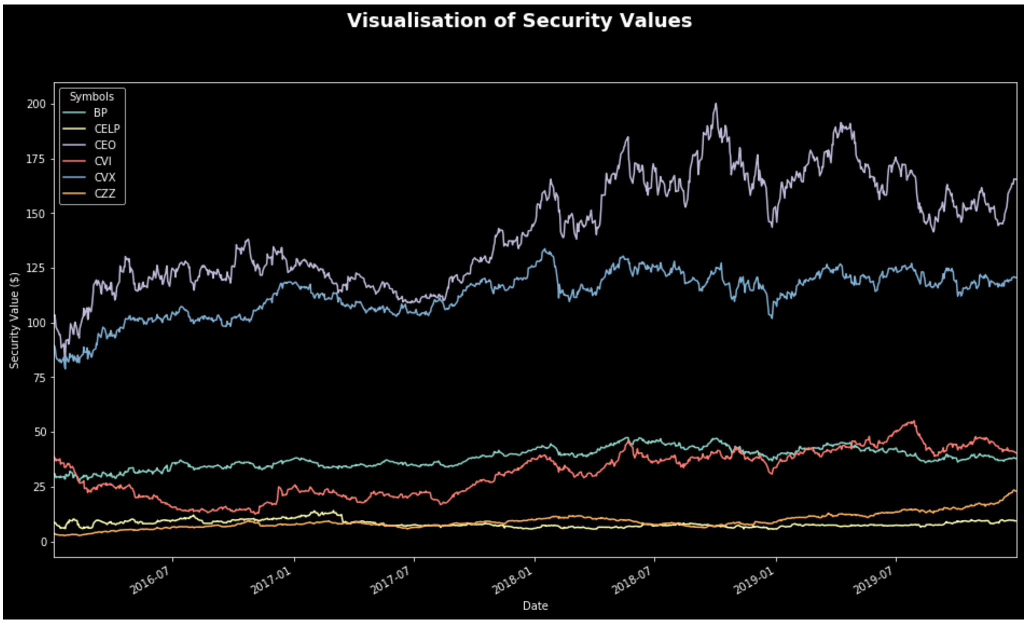


Common diagram for all securities

Additionally, the values of all the securities can be visualised in the same figure, as follows:

[6]:
```
fig = stockValues.plot(figsize=(16,9))
plt.suptitle('Visualisation of Security Values', fontsize=18,␣
 ↪fontweight='bold')
plt.ylabel("Security Value ($)")
```

[6]: Text(0, 0.5, 'Security Value ($)')



Security Returns

Returns calculation

The following step is the calculation of the arithmetical return of the securities. This step is executed by converting the pandas dataframe to numpy array in order to make the calculations and then converting the returns list back to dataframe. Given the historical values the calculation of the arithmetical return is presented below:

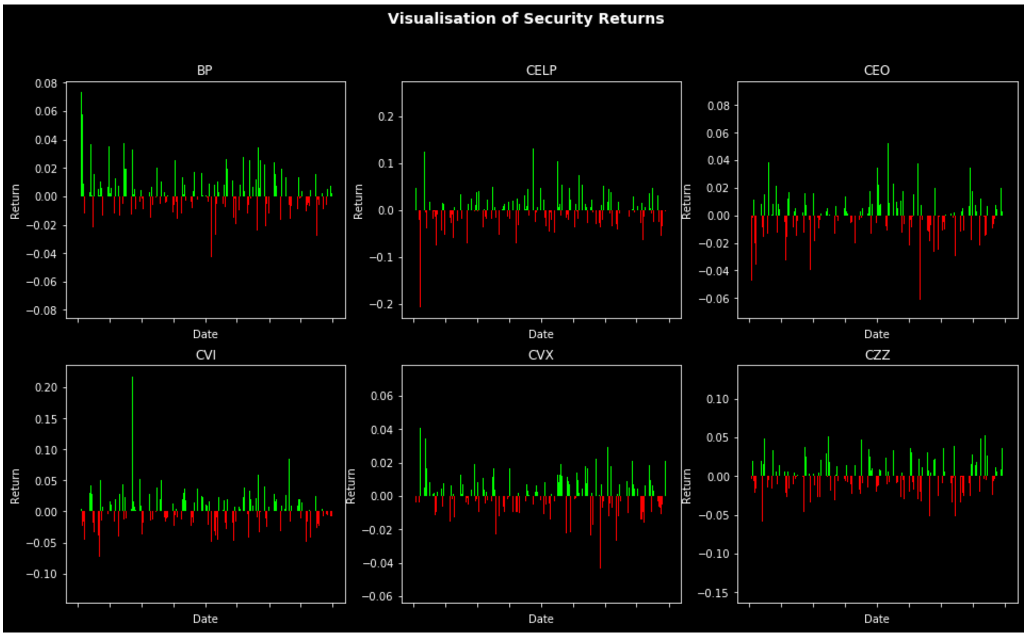
[7]:
```
stockValuesArray = pd.DataFrame(stockValues).to_numpy()
stockReturnsArray = np.empty(shape = (numOfDates-1,␣
↪numOfSecurities))
for i in range(numOfSecurities):
    for j in range(numOfDates-1):
        stockReturnsArray[j][i] =␣
↪(stockValuesArray[j+1][i]-stockValuesArray[j][i])/
↪stockValuesArray[j][i]
returnDates = stockValues.index[1:]
stockReturns = pd.DataFrame(stockReturnsArray, index=returnDates,␣
↪columns=stockValues.columns)
print("============  Stock Returns  ============ \n")
display(stockReturns)
```

```
====================  Stock Returns  ====================



Symbols              BP      CELP       CEO       CVI       CVX       CZZ
Date
2016-01-05  0.003896 -0.018182  0.004269 -0.023405 -0.005361 -0.011142
2016-01-06 -0.032018 -0.045139 -0.021739 -0.005531 -0.018080 -0.061972
2016-01-07 -0.028734 -0.018182 -0.034173 -0.027013 -0.033051 -0.072072
2016-01-08  0.011696 -0.075309 -0.005829  0.017692 -0.013720 -0.012945
2016-01-11  0.005100 -0.065421 -0.017692 -0.013373 -0.006476 -0.022951
...              ...       ...       ...       ...       ...       ...
2019-12-24  0.006895  0.001043  0.009511 -0.000967  0.009387  0.002704
2019-12-26  0.002370  0.005208 -0.001407 -0.007502  0.001993  0.044045
2019-12-27  0.004729 -0.025907  0.013907  0.000975  0.001823  0.009471
2019-12-30 -0.011506 -0.007447 -0.000786 -0.016565 -0.003722 -0.021748
2019-12-31  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000

[1005 rows x 6 columns]
```

Returns Visualisation

The visualisation of the return of each security is presented in the following figure:

[8]:
```python
fig, ax = plt.subplots(2, 3, figsize=(16,9))

for j in range(numOfSecurities):
    colors = np.array([(1,0,0)]*len(returnDates))
    for i in range(numOfDates-1):
        if stockReturnsArray[i][j] > 0:
            colors[i] = (0,1,0)
    ax[j//3,j%3].bar(returnDates, stockReturnsArray[:,j],␣
 ↪color=colors)
    ax[j//3,j%3].set_title(stockValues.columns[j])
    ax[j//3,j%3].set_xticklabels([])
    ax[j//3,j%3].set_xlabel("Date")
    ax[j//3,j%3].set_ylabel("Return")
    fig.suptitle('Visualisation of Security Returns',␣
 ↪fontweight='bold', fontsize=14)
```



Financial Statistics

Basic statistical indexes

In the next step, some fundamental statistical indexes of the data are calculated. This proccess is made with the numpy library which supports a variety of statistical calculations, as shown in the following section:

[9]:
```
from scipy.stats import kurtosis, skew

MinReturn = [0 for i in range(numOfSecurities)]
MaxReturn = [0 for i in range(numOfSecurities)]
MedianReturn = [0 for i in range(numOfSecurities)]
MeanReturn = [0 for i in range(numOfSecurities)]
SD = [0 for i in range(numOfSecurities)]
VaR99 = [0 for i in range(numOfSecurities)]
VaR97 = [0 for i in range(numOfSecurities)]
VaR95 = [0 for i in range(numOfSecurities)]
Skewness = [0 for i in range(numOfSecurities)]
Kurtosis = [0 for i in range(numOfSecurities)]
AbsMinPerSD = [0 for i in range(numOfSecurities)]

for i in range(numOfSecurities):
        MinReturn[i] = np.min(stockReturnsArray[:,i])
```

```
        MaxReturn[i] = np.max(stockReturnsArray[:,i])
        MedianReturn[i] = np.median(stockReturnsArray[:,i])
        MeanReturn[i] = np.mean(stockReturnsArray[:,i])
        SD[i] = np.std(stockReturnsArray[:,i])
        VaR99[i] = np.percentile(stockReturnsArray[:,i], 1)
        VaR97[i] = np.percentile(stockReturnsArray[:,i], 3)
        VaR95[i] = np.percentile(stockReturnsArray[:,i], 5)
        Skewness[i] = skew(stockReturnsArray[:,i], bias=False)
        Kurtosis[i] = kurtosis(stockReturnsArray[:,i], bias=False)
        AbsMinPerSD[i] = np.abs(MinReturn[i])/SD[i]

statistics = pd.DataFrame(
    {'MinReturn': MinReturn,
     'MaxReturn': MaxReturn,
     'Median': MedianReturn,
     'Mean': MeanReturn,
     'SD': SD,
     'VaR99': VaR99,
     'VaR97': VaR97,
     'VaR95': VaR95,
     'Skewness': Skewness,
     'Kurtosis': Kurtosis,
     'AbsMinPerSD': AbsMinPerSD,
    }, index=stockValues.columns)


display(statistics)
```

```
         MinReturn  MaxReturn    Median      Mean        SD     VaR99  \
Symbols
BP       -0.078008   0.073658  0.000274  0.000311  0.014651 -0.037868
CELP     -0.207407   0.252144  0.000000  0.000704  0.036102 -0.088097
CEO      -0.066644   0.089765  0.000671  0.000620  0.017273 -0.046547
CVI      -0.129289   0.217920  0.000633  0.000386  0.026544 -0.062649
CVX      -0.057246   0.071827  0.000417  0.000376  0.012722 -0.030420
CZZ      -0.149343   0.129114  0.001453  0.002143  0.024357 -0.060368

            VaR97     VaR95  Skewness  Kurtosis  AbsMinPerSD
Symbols
BP      -0.028095 -0.023373 -0.061069  2.605570     5.324494
CELP    -0.067235 -0.055164  0.569211  6.275317     5.745003
CEO     -0.031514 -0.026274  0.047546  1.686140     3.858304
CVI     -0.048366 -0.042152  0.700962  7.239242     4.870644
CVX     -0.024094 -0.020710 -0.017391  2.519382     4.499711
CZZ     -0.039525 -0.033042  0.050106  3.464028     6.131390
```

Covariance - Correlation

Now, given the arithmetical returns of the securities we can compute the variance-covariance matrix among all the securities. The computation can be achieved with the pandas function *cov()* which calculates the covariance matrix of a dataframe.

[10]:
```
cov = stockReturns.cov()
covarianceMatrix = np.array(cov)
print("============  Covariance Matrix  ============")
display(cov)
```

```
=====================  Covariance Matrix  =====================
```

Symbols	BP	CELP	CEO	CVI	CVX	CZZ
Symbols						
BP	0.000215	0.000081	0.000160	0.000119	0.000119	0.000123
CELP	0.000081	0.001305	0.000119	0.000106	0.000073	0.000140
CEO	0.000160	0.000119	0.000299	0.000130	0.000118	0.000152
CVI	0.000119	0.000106	0.000130	0.000705	0.000105	0.000133
CVX	0.000119	0.000073	0.000118	0.000105	0.000162	0.000098
CZZ	0.000123	0.000140	0.000152	0.000133	0.000098	0.000594

The correlation matrix can be computed accordingly, with the *corr()* function:

[11]:
```
correlation = stockReturns.corr()
print("============  Correlation Matrix  ============")
correlation.style.background_gradient(cmap='Wistia').
  ↪set_precision(4)
```

Portfolio Optimisation

In this section we attempt to optimise the portfolio of securities, using the mean - variance method. The optimisation problem is a quadratic bi-objective problem, which will be solved parametrically setting the expected return as a parameter. We use the scipy optimiser library in order to solve the problem:

Global Minimum Variance Portfolio

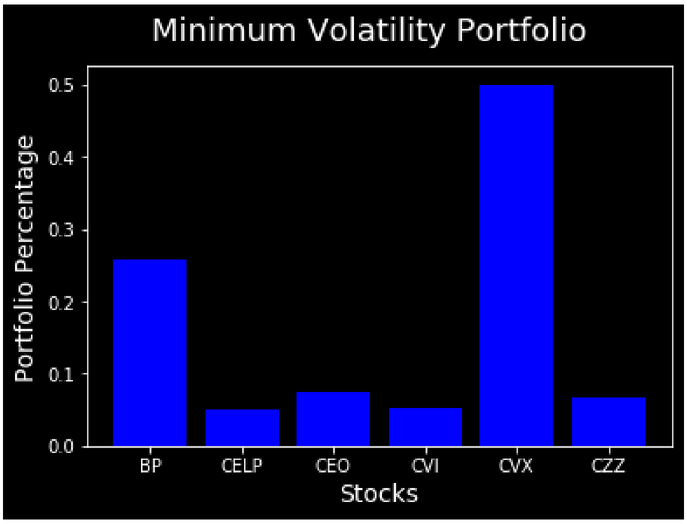
Initially, we compute the global minimum variance portfolio (GMVP) using the scipy minimise function. The minimise function takes as an argument the mean return column-vector and the covariance 2D matrix and computes the proportions of the GMVP minimising the quantity defined in the function named 'Portfolio Volatility' which is the standard deviation of the portfolio. Additionally, we set the constraint that the weights sum to 1 and that the bounds of the proportions are (0,1), imposing the short sales restriction.

[12]:
```
#Objective Function
def portfolioVolatility(weights, MeanReturn, covarianceMatrix):
    std = np.sqrt(np.dot(weights.T, np.dot(covarianceMatrix,␣
 ↪weights)))
    return std

#Constraints
args = (MeanReturn, covarianceMatrix)
constraints = ({'type': 'eq', 'fun': lambda x: np.sum(x) - 1})
bound = (0,0.5)
bounds = tuple(bound for asset in range(numOfSecurities))

#Optimisation Function
minVolatilityPortfolio = sco.minimize(portfolioVolatility,␣
 ↪numOfSecurities*[1./numOfSecurities,], args=args,␣
 ↪method='SLSQP', bounds=bounds, constraints=constraints)

sdPort1 = np.sqrt(np.dot(minVolatilityPortfolio['x'].T, np.
 ↪dot(covarianceMatrix, minVolatilityPortfolio['x'])))
retPort1 = np.sum(MeanReturn*minVolatilityPortfolio['x'] )


print("Risk of minimum volatility portfolio:", sdPort1)
print("Return of minimum volatility portfolio:", retPort1)
print("Sharpe Ratio of minimum volatility portfolio:", retPort1/
 ↪sdPort1)

fig = plt.figure()
plt.bar(tickers,minVolatilityPortfolio['x'], color='b')
fig.suptitle('Minimum Volatility Portfolio', color='white',␣
 ↪size='18')
plt.xlabel('Stocks', size='14')
plt.ylabel('Portfolio Percentage', size='14')
```

```
Risk of minimum volatility portfolio: 0.01185387330551697
```

```
Return of minimum volatility portfolio: 0.0005110703485548262
Sharpe Ratio of minimum volatility portfolio: 0.04311420709355536
```

[12]: Text(0, 0.5, 'Portfolio Percentage')



Max Sharpe Ratio Portfolio

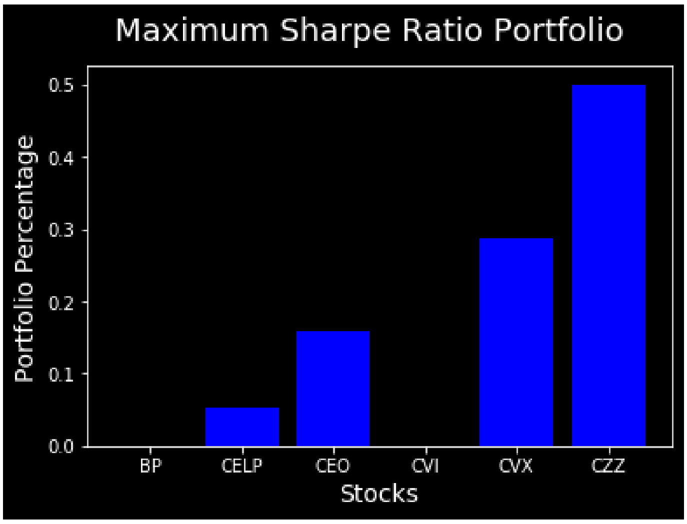
With the same function we can compute the portfolio that maximises the sharpe ratio. The only difference between this and the previous step is the different minimisation function, which now is altered in order to maximise sharpe ratio. Because of the definition of the scipy minimise function we should form a minimisation problem. Therefore, we define the quantity of negative sharpe ratio, which should be minimised in order to maximise positive sharpe ratio.

[13]:
```
#Objective Function
def negSharpeRatio(weights, MeanReturn, covarianceMatrix):
    returns = np.sum(MeanReturn*weights )
    std = np.sqrt(np.dot(weights.T, np.dot(covarianceMatrix,␣
 ↪weights)))
    return (- returns / std)

#Constraints
args = (MeanReturn, covarianceMatrix)
constraints = ({'type': 'eq', 'fun': lambda x: np.sum(x) - 1})
bound = (0,0.5)
```

```
bounds = tuple(bound for asset in range(numOfSecurities))

#Optimisation Function
maxSharpeRatioPortfolio = sco.minimize(negSharpeRatio,␣
↪numOfSecurities*[1./numOfSecurities,], args=args,␣
↪method='SLSQP', bounds=bounds, constraints=constraints)

sdPort2 = np.sqrt(np.dot(maxSharpeRatioPortfolio['x'].T, np.
↪dot(covarianceMatrix, maxSharpeRatioPortfolio['x'])))
retPort2 = np.sum(MeanReturn*maxSharpeRatioPortfolio['x'] )

print("Risk of maximum sharpe ratio portfolio:", sdPort2)
print("Return of maximum sharpe ratio portfolio:", retPort2)
print("Maximum Sharpe Ratio:", retPort2/sdPort2)

fig = plt.figure()
plt.bar(tickers,maxSharpeRatioPortfolio['x'], color='b')
fig.suptitle('Maximum Sharpe Ratio Portfolio', color='white',␣
↪size='18')
plt.xlabel('Stocks', size='14')
plt.ylabel('Portfolio Percentage', size='14')
```

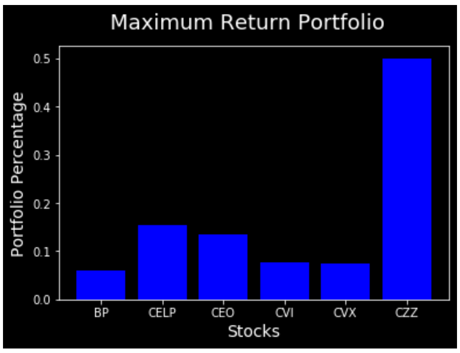
```
Risk of maximum sharpe ratio portfolio: 0.015744182005376268
Return of maximum sharpe ratio portfolio: 0.0013158973803388851
Maximum Sharpe Ratio: 0.0835799141479375
```

[13]: Text(0, 0.5, 'Portfolio Percentage')



[14]:
```
#Objective Function
def negReturn(weights, MeanReturn, covarianceMatrix):
    returns = np.sum(MeanReturn*weights )
    return (- returns)

#Constraints
args = (MeanReturn, covarianceMatrix)
constraints = ({'type': 'eq', 'fun': lambda x: np.sum(x) - 1})
bound = (0,0.5)
bounds = tuple(bound for asset in range(numOfSecurities))

#Optimisation Function
maxReturnPortfolio = sco.minimize(negReturn, numOfSecurities*[1./
 ↪numOfSecurities,], args=args, method='SLSQP', bounds=bounds,␣
 ↪constraints=constraints)

sdPort3 = np.sqrt(np.dot(maxReturnPortfolio['x'].T, np.
 ↪dot(covarianceMatrix, maxReturnPortfolio['x'])))
retPort3 = np.sum(MeanReturn*maxReturnPortfolio['x'] )

print("Risk of maximum return portfolio:", sdPort3)
print("Maximum Return:", retPort3)
print("Sharpe Ratio of maximum return portfolio:", retPort3/
 ↪sdPort3)

fig = plt.figure()
plt.bar(tickers,maxReturnPortfolio['x'], color='b')
fig.suptitle('Maximum Return Portfolio', color='white', size='18')
plt.xlabel('Stocks', size='14')
plt.ylabel('Portfolio Percentage', size='14')
```

```
Risk of maximum return portfolio: 0.016711916652493088
Maximum Return: 0.0013399086900049058
Sharpe Ratio of maximum return portfolio: 0.08017684134422834
```

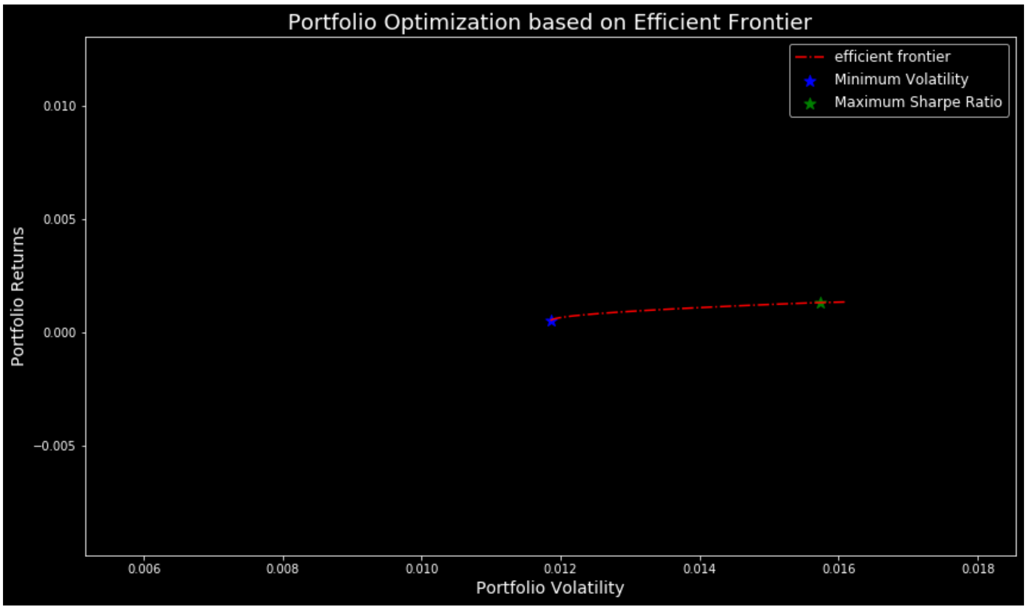


Efficient Frontier

In this step, we parametrically solve the same problem in order to gradually find the efficient frontier. Therefore, we compute a number of efficient portfolios between the GMVP and the maximum return portfolio.

[15]:
```
numOfPortfolios = 20
returnRange = np.linspace(retPort1, retPort3, numOfPortfolios)
efficientFrontier = []
AllReturns = []
AllSDs = []
for target in returnRange:
    args = (MeanReturn, covarianceMatrix)
    constraints = ({'type': 'eq', 'fun': lambda x: np.
 ↪sum(MeanReturn*x) - target},
                       {'type': 'eq', 'fun': lambda x: np.
 ↪sum(x) - 1})
    bounds = tuple((0,0.5) for asset in range(numOfSecurities))
    result = sco.minimize(portfolioVolatility, numOfSecurities*[1./
 ↪numOfSecurities,], args=args, method='SLSQP', bounds=bounds,␣
 ↪constraints=constraints)
    efficientFrontier.append(result)
    AllSDs.append(np.sqrt(np.dot(result['x'].T, np.
 ↪dot(covarianceMatrix, result['x']))))
    AllReturns.append(np.sum(MeanReturn*result['x']))

fig = plt.figure(figsize=(14, 8))
ax = plt.subplot(1,1,1)
plt.scatter(sdPort1, retPort1, c='b', marker='*', s=100,␣
 ↪label='Minimum Volatility')
plt.scatter(sdPort2, retPort2, c='g', marker='*', s=100,␣
 ↪label='Maximum Sharpe Ratio')
plt.plot([p['fun'] for p in efficientFrontier], AllReturns,␣
 ↪linestyle='-.', color='red', label='efficient frontier')
ax.set_title('Portfolio Optimization based on Efficient Frontier',␣
 ↪color='white', size='18')
plt.xlabel('Portfolio Volatility', size='14')
plt.ylabel('Portfolio Returns', size='14')
plt.legend(prop={'size': 12})
```

[15]:
```
<matplotlib.legend.Legend at 0x7f1f488ffb90>
```

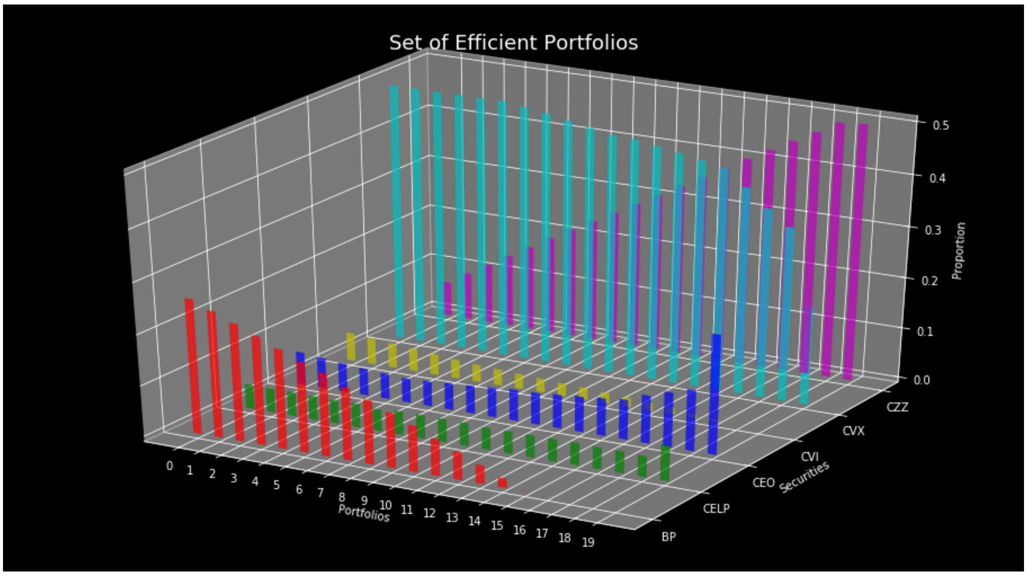


The formulated portfolios are presented below:

[16]:
```
efficientPortfolios = [0 for i in range(numOfPortfolios)]
for i in range(numOfPortfolios):
    efficientPortfolios[i] = efficientFrontier[i].x

weightingFactor = [[0 for i in range(numOfPortfolios)] for j in␣
 ↪range(numOfSecurities)]
for i in range(numOfSecurities):
    for j in range(numOfPortfolios):
        weightingFactor[i][j] = efficientFrontier[j].x[i]

from mpl_toolkits.mplot3d import Axes3D

fig = plt.figure(figsize=(16,9))
ax = fig.add_subplot(111, projection='3d')
for c, z in zip(['r', 'g', 'b', 'y', 'c', 'm'], [0, 1, 2, 3, 4,␣
 ↪5]):
    xs = np.arange(numOfPortfolios)
    ys = weightingFactor[z]
    cs = [c] * len(xs)
    ax.bar(xs, ys, zs=z, zdir='y', color=cs, alpha=0.7, width=0.4)
plt.yticks(np.arange(6), tickers)
plt.xticks(np.arange(numOfPortfolios))

ax.set_xlabel('Portfolios')
ax.set_ylabel('Securities')
```

```
ax.set_zlabel('Proportion')
ax.set_title('Set of Efficient Portfolios', color='white',␣
 ↪size='18')
```

[16]: Text(0.5, 0.92, 'Set of Efficient Portfolios')



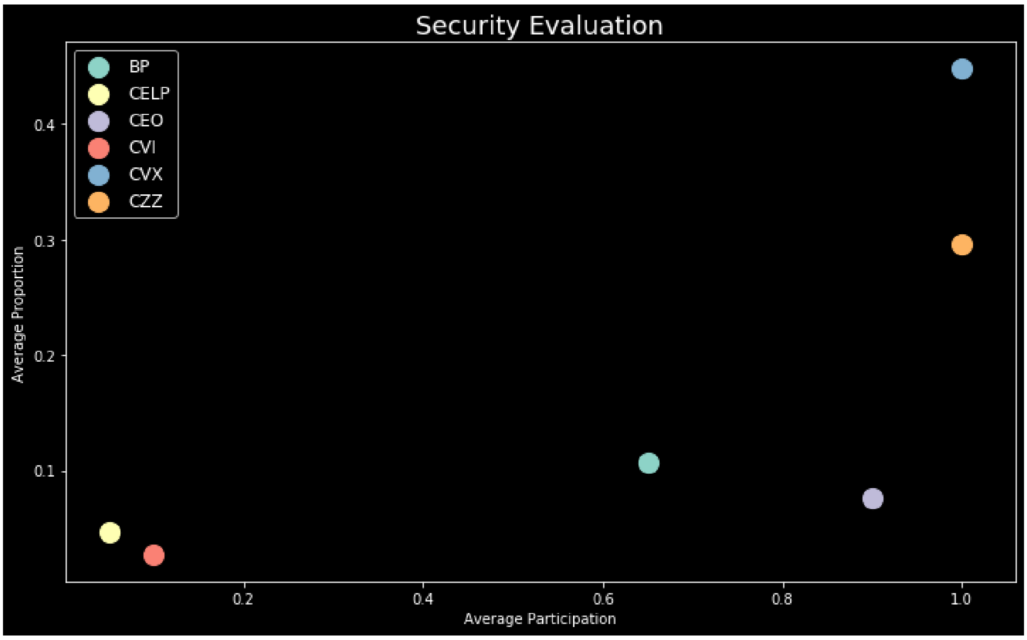
Security average proportion and participation

Finally, we can determine the percrentage of participation of each security in the efficient portfolios in order to obtain another perspective of the solution. In the following section, we present the according idea:

[17]:
```
securityParticipation = [0 for i in range(numOfSecurities)]
for i in range(numOfSecurities):
    for j in range(numOfPortfolios):
        if weightingFactor[i][j] >= 0.05:
            securityParticipation[i] = securityParticipation[i] + 1
    securityParticipation[i] = securityParticipation[i] /␣
 ↪numOfPortfolios

securityAvgProportion = [0 for i in range(numOfSecurities)]
for i in range(numOfSecurities):
    securityAvgProportion[i] = np.mean(weightingFactor[i])

fig = plt.figure(figsize=(12,7))
```

```
ax = fig.add_subplot(111)

for i in range(numOfSecurities):
    plt.scatter(securityParticipation[i],␣
 ↪securityAvgProportion[i], label=tickers[i], s=200)
plt.legend(prop={'size': 12})
ax.set_xlabel('Average Participation')
ax.set_ylabel('Average Proportion')
ax.set_title('Security Evaluation', color='white', size='18')
```

[17]: Text(0.5, 1.0, 'Security Evaluation')



The securities which are placed in the upper right section of the figure are the most dominant investment opportunities.

Chapter 6

Empirical Testing

6.1 Introduction

This chapter presents the empirical testing of the proposed methodology on real data. This step is very important, because it is necessary that the methodology is explicitly tested to verify its validity. Consequently, a large experimental application of the proposed methodological framework was conducted, including securities from four stock exchanges (NYSE, NASDAQ, Paris and Tokyo). Thus, every step of the methodological framework is examined, while the results are tested with an out-of-sample validation process.

The input data for the first phase of the process (incl. financial indexes of the securities) were drawn from `www.investing.com`. The input data for the second phase of the process (incl. historical data for time-horizon of 3 years) were drawn from the `www.finance.yahoo.com`.

In the second section of the chapter, we present the main characteristics of the application field. The third section of the chapter includes a detailed description of the empirical testing process for NYSE stock exchange. The corresponding procedure for the other three stock exchanges (NASDAQ, Paris and Tokyo) is described in appendix B. This section is divided into two parts. The first part describes the first phase of the methodology while the second part presents the results of the second phase of the methodology. Finally, the fourth section contains the out-of-sample validation procedure of the results.

6.2 Empirical Testing Information

The proposed methodology was applied to four stock exchanges: (a) *NYSE*, (b) *NASDAQ*, (c) *Paris* and (d) *Tokyo*. The total number of the examined securities is about two thousands, while the time horizon of the analysis was set to four calendar years.

The set of the examined securities was splited according to the industrial sector and the stock exchange of each security. The companies which participated in the empirical testing process belong to three industrial sectors: (a) *technological*, (b) *energy* and (c) *financial*. The input data for the first phase were fetched from the `www.investing.com` database. However, for a large number of securities there were insufficient data. Therefore, the companies that did not satisfy the requirements (missing data, zero values etc.) were excluded from the experiment. In table 6.1 we record the securities of each stock exchange, splitted according to their industrial sector. In the last column we present the total number of securities in each sector (including the securities with insufficient data).

Stock exchange	Industrial sector	Number of experiment securities	Number of securities with missing data	Total number of securities
NYSE	technology	69	177	246
	energy	89	131	220
	financial	358	461	819
NASDAQ	technology	326	213	539
	energy	6	40	46
	financial	93	471	564
Paris	technology	50	91	141
	energy	7	8	15
	financial	33	24	57
Tokyo	technology	485	263	748
	energy	30	4	34
	financial	143	51	194

Table 6.1: Empirical testing input information

In tables 6.2 - 6.5, there is a presentation of the securities of each stock exchange. More specifically, the securities of NYSE stock exchange are recorded in table 6.2, the securities of NASDAQ stock exchange are recorded in table 6.3, the securities of Paris stock exchange are recorded in table 6.4 and, finally, the securities of Tokyo stock exchange are recorded in table 6.5.

These tables include only the securities which were used in the empirical testing process (companies with missing data are not recorded). The first column includes the securities of the technological sector, the second column includes the securities of the energy sector and the third column includes the securities of the financial sector.

	Technology	**Energy**	**Financial**
1	ABB ADR	Petroleo Brasileiro ADR Reptg 2 Pref	Nuveen CA MVF 2
2	Accenture	Phillips 66	Nuveen High Income 2020 Target Term
3	SAP ADR	Phillips 66 Partners LP	Nuveen Dow 30Sm
4	Infosys ADR	Baker Hughes A	Ellsworth Growth Pref A
5	Wipro ADR	GasLog Partners Pref A	Federal Agricultural Mortgage A
6	BT ADR	Adams Resources&Energy	Chimera Investment Pref A
7	STMicroelectronics ADR	Ecopetrol ADR	Ares Management Pref A
8	Canon ADR	Total ADR	Apollo Global Management A
9	Agilent Technologies	Petroleo Brasileiro Petrobras ADR	Ladder Capital A
10	Allegion PLC	CNOOC ADR	Aberdeen Emerging Markets Equity
11	Ametek	Sinopec Shanghai Petrochemical ADR	Aberdeen Asia-Pacific
12	Amphenol	Royal Dutch Shell ADR	Adams Diversified Equity Closed
13	AO Smith	Equinor ADR	Barclays ADR
14	Scnc App In	ENI ADR	Santander Chile ADR
15	Rockwell Automation	PetroChina ADR	Sumitomo Mitsui Financial ADR
16	AVX	Transportadora Gas ADR	Mitsubishi UFJ Financial ADR
17	AZZ	BP ADR	China Life Insurance ADR
18	Badger Meter	Royal Dutch Shell B ADR	Aegon ADR
19	Belden	Plains All American Pipeline	Banco Bilbao ADR
20	Regal Beloit	YPF Sociedad Anonima	Credit Suisse ADR
21	Benchmark Electronics	Archrock	Prudential Public ADR
22	Broadridge	Teck Resources B	Lloyds Banking ADR
23	BWX Tech	BP Prudhoe Bay Royalty Trust	ING ADR
24	CAE Inc.	Cabot Oil&Gas	BBVA Banco Frances ADR
25	Jabil Circuit	Canadian Natural	Santander ADR
26	TE Connectivity	Cenovus Energy Inc	Itau CorpBanca ADR
27	Issuer Direct Corp	Chevron	Westpac Banking ADR
28	CTS Corp	Cimarex Energy	Nuveen California Div Advantag Muni
29	Danaher	CONSOL Coal	BlackRock Long Term Muni Advantage
30	Deluxe	Concho Resources	Aflac
31	DXC Technology	ConocoPhillips	AG Mortgage Investment
32	Eaton	Continental Resources	AG Mortgage Invest Trust Pb Pref
33	Espey Mfg&Electronics	Cosan Ltd	AG Mortgage Invest Trust Pa Pref
34	Methode Electronics	Crestwood Equity Partners LP	Federal Agricultural Mortgage
35	Emerson	Crossamerica Partners LP	Great Ajax Corp
36	Energizer	CVR Energy	Alliance Data Systems
37	Enersys	Cypress Energy Partners LP	AllianceBernstein Holding LP
38	ESCO Technologies	Delek Logistics Partners LP	AllianzGI Diversifiedome Convertibl
39	Evertec Inc	Delek US Energy	AllianzGI Equity Convertible Closed
40	FactSet Research	Devon Energy	Ares Dynamic Credit Allocation Inc
...	...	...	...
69	Xerox	MPLX LP	Arthur J Gallagher
70		Suburban Propane Partners LP	Western Asset Mortgage
...		...	...
89		Williams	Great Western Bancorp Inc
90			Berkshire Hills Bancorp
...			...
358			Westwood

Table 6.2: List of NYSE securities used in empirical testing

	Technology	**Energy**	**Financial**
1	Bel Fuse A	Diamondback	Nuveen NASDAQ 100 Dyn Over
2	Cognizant A	Alliance Resource	1st Source
3	Activision Blizzard	Viper Energy Ut	1st Constitution Bancorp
4	Formula Systems ADR	Dorchester Minerals	Bancorp 34
5	LM Ericsson B ADR	Hallador	National General A Pref
6	Allied Motion	TransGlobe Energy	Donegal A
7	Amdocs		ACNB
8	American Software		Hennessy Ad
9	Analog Devices		Grupo Financiero Galicia ADR
10	Apple		Alcentra Capital Corp
11	Applied Materials		Alerus Fin
12	Jack Henry&Associates		Amark Preci
13	AstroNova		America First Tax
14	AudioCodes		German American Bancorp
15	Hollysys Automation Tech		American National Bankshares
16	Avnet		American River
17	Bel Fuse B		Atlantic American
18	Blackbaud		American National Insurance
19	Broadcom		Ameris
20	Bruker		AMERISAFE
21	Cabot		AmeriServ
22	Camtek		TD Ameritrade
23	CDK Global Holdings LLC		Ames
24	CDW Corp		Apollo Invest
25	Cerner		Ares Capital
...	...		...
93	Xperi		Northwest Bancshares
94			Huntington Bancshares
...			...
329			Zions

Table 6.3: List of NASDAQ securities used in empirical testing

	Technology	**Energy**	**Financial**
1	Akka	Total	BNP Paribas
2	Alten	TechnipFMC	AXA
3	Artois Nom.	Rubis	Credit Agricole
4	Atos	GTT	Societe Generale
5	Aubay	Total Gabon	Amundi
6	Aures Tech	Maurel et Prom	Natixis
7	Axway	Docks des Petroles dAmbes	CNP Assurances
8	Capgemini		SCOR
9	Cofidur		Euronext
10	Coheris		Eurazeo
11	CS Communication		FFP
12	Dassault Systemes		Rothschild & Co
13	Delfingen		CRCAM Langued
14	Devoteam		CRCAM Brie Picardie 2
15	DNXcorp		Coface
16	Schneider Electric		CRCAM Atlantique
17	Environnement		April
18	Esker		Crcam Touraine
19	Evolis		Crcam Ille-Vil
20	Fiducial Office		Altamir
21	GEA		Ca Toulouse 31 CCI
22	Perrier Gerard		Crcam Morbihan
23	ITS Group		Galimmo
24	Groupe Open		ABC Arbitrage
25	Guillemot		Viel Et Compagnie
26	Harvest		Union Financiere
27	Hitechpros		IDI
28	Infotel		Crcam Norm.Sei
29	Ingenico		Crcam Sud RA
30	Innelec		Lebon
31	Pharmagest Interactive		Crcam Loire Ht
32	Lacroix		Groupe IRD
33	Legrand		Idsud
34	Linedata Services		
35	Mersen		
36	Neurones		
37	Prodware		
38	Quadient		
39	Rexel		

Table 6.4: List of Paris Stock Exchange securities used in empirical testing

	Technology	Energy	Financial
1	Yaskawa Electric Corp.	San-Ai Oil	The 77 Bank Ltd
2	Advantest Corp.	BP Castrol KK	Nihon M&A Center
3	Rohm Ltd	Mitsui Matsushima Co Ltd	Acom Co Ltd
4	Hitachi High-Technologies Corp	Idemitsu Kosan Co Ltd	Activia Properties
5	Nitto Denko Co	Sinanen Co Ltd	MS&AD Insurance Group Holdings
6	Shimadzu Corp	Itochu Enex Co Ltd	Advance Create
7	Otsuka Corp	Toell Co Ltd	Japan Investment Adviser
8	Disco Corp	Nippon Coke & Engineering Ltd	Aeon Financial Service Co Ltd
9	Trend Micro Inc.	Inpex Corp.	Aichi Bank Ltd
10	Ricoh	Marubeni Corp.	Aizawa Securities
11	Konami Corp.	Sojitz Corp.	Akatsuki
12	Itochu Techno Solutions	Sala Corp	Akita Bank Ltd
13	Hamamatsu Photonics KK	Kamei Corp	Anicom Holdings Inc
14	It Holdings Corp	Iwatani Corp	Anshin Guarantor Service
15	SCSK Corp	Tokai Holdings Corp	Aomori Bank Ltd
16	Nikon Corp.	MORESCO Corp	Asax Co Ltd
17	Brother Industries Ltd	Cosmo Energy Holdings	Ashikaga Holdings
18	Seiko Epson Cor	Daimaru Enawin	Astmax
19	KakakuCom Inc	K&O Energy Group Inc	Awa Bank Ltd
20	NGK Insulators	JP Petroleum Exploration Ltd	Yamanashi Chuo Bank
21	Alps Electric	Yamashin-Filter	Shiga Bank Ltd
22	Hirose Electric Co Ltd	Mitsuuroko Group Holdings	Kiyo Bank Ltd
23	Yokogawa Electric Corp.	Sumiseki Holdings Inc	Bank of Kochi Ltd
24	Fuji Electric	JX Holdings, Inc.	Chukyo Bank Ltd
25	SUMCO Corp.	Taiheiyo Kouhatsuorporated	Taiko Bank Ltd
26	Azbil Corp	Mitsui	Kita Nippon Bank
27	Konica Minolta, Inc.	Nissin Shoji	Yamagata Bank Ltd
28	Nihon Unisys Ltd	Toa Oil	Oita Bank Ltd
29	Lasertec Corp	Shinko Plantech	Miyazaki Bank Ltd
30	Taiyo Yuden	Sanrin	Ehime Bank Ltd
31	Ns Solutions Corp		Fukushima Bank Ltd
32	Ibiden Co Ltd		The Bank Of Kyoto Ltd
33	Dainippon Screen Mfg.		The Chugoku Bank Ltd
34	Obic Business Consultants		The Iyo Bank Ltd
35	Capcom Co Ltd		The Hiroshima Bank Ltd
36	Koei Tecmo Holdings		Chiba Kogyo Bank
37	Canon Marketing Japan Inc		Bank of Iwate Ltd
38	DeNA Co		Michinoku Bank Ltd
39	Anritsu Corp		San-in Godo Bank
...	...		...
142	FTGroup		United Urban
143	Marvelous Inc		
...	...		
482	Obic Co Ltd		

Table 6.5: List of Tokyo stock exchange securities used in empirical testing

6.3 Results Presentation

In this section there is a detailed presentation of a set of tables and figures that describe the input data and the obtained results of the empirical testing process during each phase of the methodological framework.

It is necessary to mark the fact that this section refers to the NYSE stock exchange experimental results. The results for the remaining three stock exchanges are placed in Appendix B of the thesis.

The first part of the section includes the input data of the empirical testing. In the second paragraph, the results of the first phase of the methodological framework are presented. Finally, in the third paragraph, the results of the second phase of the methodology are presented.

6.3.1 Phase I: Multicriteria portfolio selection

The first phase of the empirical testing includes the portfolio selection process, based on multicriteria decision analysis methods. The aim of this phase is to locate the securities which have a strong evolutionary potential. The process is based on four multicriteria decision analysis methods.

The input values that must be determined in order to perform the four ranking methods include the *evaluation matrix* which contains the performance of each alternative in the determined financial criteria, as well as the *offsets* and the *thresholds* for each criterion.

The selection of the offsets was determined according to three different scenarios, in order to conduct a sensitivity analysis on the results. In this paragraph, the results according to the scenario of equal offsets among the alternatives are presented.

The thresholds configuration differs significantly according to the multicriteria method. For each ranking method the configuration process was based on the particion of the values' range. Firstly, we determine the range by calculating the minimum and maximum value of the alternatives for each criterion and secondly we split this range as follows: (i) For ELECTRE III method which invloves three different thresholds (preference p, indifference q and veto v) we split the range in four sections and assign the values respectively: $q(i) < p(i) < v(i)$, (ii) For PROMETHEE II method which invloves two different thresholds (preference p and indifference q) we split the range in three sections and assign the values respectively: $q(i) < p(i)$, (iii) MAUT and TOPSIS methods do not involve any thresholds.

In tables 6.6 - 6.8 the input data for each industrial sector are presented. More specifically, the companies of the energy sector are recorded in table 6.6, the companies of the financial sector are recorded in table 6.7 and finally the companies from the technological sector are recorded in table 6.8.

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Petroleo Brasileiro ADR Reptg 2 Pref	12.23	1.15	101.69	1.47	1.61	3	11.56	-6.44
Phillips 66	9.21	11.57	109.68	1.07	3.38	5	23.74	-2.61
Phillips 66 Partners LP	13.26	4.16	1.1	0.86	6.19	5	31.09	7.52
Baker Hughes A	170.91	0.13	23.54	1.02	3.3	1	1.4	-29.34
GasLog Partners Pref A	10.31	1.91	0.376	0.98	11.17	5	16.76	0.59
Adams Resources&Energy	40.77	0.73	1.84	0.78	3.21	1	-22.76	-25.88
Ecopetrol ADR	10.41	1.64	21.64	1.6	8.14	1	7.43	-33.1
Total ADR	12.39	3.96	185.04	0.75	5.92	1	-4.48	-19.06
Petroleo Brasileiro Petrobras ADR	12.25	1.15	101.69	1.47	1.61	2	8.3	-8.21
CNOOC ADR	8.24	18.09	34.02	1.07	6.25	1	-2.22	-18.59
Sinopec Shanghai Petrochemical ADR	7.57	3.8	14.14	0.99	12.63	1	-33.32	-41.24
Royal Dutch Shell ADR	11.58	4.94	376.66	0.87	6.57	1	-1.73	-12.26
Equinor ADR	7.49	2.47	74.02	0.99	5.3	1	-12.71	-31.25
ENI ADR	14.31	2.1	128.73	0.78	6.16	1	-4.44	-16.71
PetroChina ADR	12.39	4.14	360.8	1.16	7.21	1	-16.62	-31.94
Transportadora Gas ADR	4.57	1.75	0.952	0.89	22.68	1	-44.67	-39.68
BP ADR	14.13	2.63	294.14	0.77	6.62	1	-2	-16.59
Royal Dutch Shell B ADR	11.58	4.94	376.66	0.87	6.57	1	-3.64	-14.19
Plains All American Pipeline	4.87	3.94	34.21	1.02	7.51	1	-4.39	-21.38
YPF Sociedad Anonima	8.36	1.1	12.49	1.46	2.37	1	-29.05	-35.55
Archrock	22.73	0.41	0.94	2.85	6.18	1	25.37	-17.78
Teck Resources B	4.45	3.34	9.61	1.51	1.02	1	-26.37	-31.73
BP Prudhoe Bay Royalty Trust	2.07	4.14	0.089	-0.2	26.03	1	-60.44	-76.49
Cabot Oil&Gas	9.08	1.93	2.44	0.54	2.06	1	-21.66	-25.11
Canadian Natural	8.28	3	15.45	1.21	3.96	1	3.11	-14.33
Cenovus Energy Inc	44.31	0.18	15.9	1.02	1.43	1	16.22	-6.74
Chevron	14.68	7.71	152.89	1.01	4.21	2	5.33	-2.7
Cimarex Energy	7.24	6.19	2.34	1.36	1.78	1	-27.23	-51.78
CONSOL Coal	7.2	1.75	0.34	1.06	16.26	1	-23.71	-30.94
Concho Resources	26.24	2.45	4.49	1.26	0.78	1	-37.55	-57.33
ConocoPhillips	8.88	6.18	35.94	1.05	3.06	1	-10.38	-24.31
Continental Resources	11.11	2.5	4.76	1.72	0.72	1	-30.93	-54.08
Cosan Ltd	12.35	1.22	5.78	1.19	0.55	5	70.45	107.76
Crestwood Equity Partners LP	13.75	2.55	3.22	2.04	6.85	3	25.47	-5.25
...	...	...	...	...	...	...	...	...
Williams	327.38	0.07	8.6	1.56	6.68	1	3.13	-14.77

Table 6.6: Evaluation Table input data for NYSE Energy Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Nuveen CA MVF 2	43.47	0.38	0.002	0.06	3.09	5	8.01	4.77
Nuveen High Income 2020 Target Term	18.47	0.54	0.01	0.02	3.19	2	-0.4	1.54
Nuveen Dow 30Sm	15.72	1.12	0.015	0.97	6.72	3	9.12	-4.71
Ellsworth Growth Pref A	9.21	1.15	0.003	0.88	4.54	5	11.69	6.43
Federal Agricultural Mortgage A	9.16	8.94	0.609	1.19	3.42	5	26.23	10.68
Chimera Investment Pref A	21.89	0.91	1.36	0.62	10.06	5	6.26	5.08
Ares Management Pref A	39.19	0.68	1.35	1.47	4.64	5	4.65	5.18
Apollo Global Management A	30.02	1.29	1.72	1.49	5.12	5	60.76	29.9
Ladder Capital A	12.6	1.36	0.553	1.03	7.94	5	10.73	-2
Aberdeen Emerging Markets Equity	10.31	0.68	0.02	0.71	3.08	3	11.34	10.3
Aberdeen Asia-Pacific	71.33	0.06	0.083	0.51	7.8	3	9.3	5.49
Adams Diversified Equity Closed	7.95	1.99	0.031	1	1.27	5	25.67	1.28
Barclays ADR	9.14	0.91	26.52	0.97	4.28	1	3.9	-16.36
Santander Chile ADR	16.67	1.73	2.66	0.63	3.83	1	6.76	-11.69
Sumitomo Mitsui Financial ADR	7.37	0.94	24.33	1.23	9.39	3	-3.08	-4.48
Mitsubishi UFJ Financial ADR	8.26	0.61	29.95	1.37	3.94	1	10.74	-1.76
China Life Insurance ADR	14.81	0.81	101.14	1.54	0.97	1	-8.39	-29
Aegon ADR	10.19	0.41	49.17	1.36	7.94	1	14.68	11.91
Banco Bilbao ADR	6.65	0.79	24.59	1.06	5.52	1	0.38	-8.61
Credit Suisse ADR	13.51	0.91	18.39	1.44	2.09	1	13.44	-6.31
Prudential Public ADR	20.99	1.77	65.73	1.46	3.4	1	-62.49	-57.29
Lloyds Banking ADR	11.21	0.27	54.24	1.06	7.13	1	6.19	-6.06
ING ADR	8.35	1.3	24.65	1.35	6.98	1	5.03	-6.26
BBVA Banco Frances ADR	2.69	1.58	1.45	0.73	6.13	2	20.31	5.84
Santander ADR	8.17	0.5	67.96	1.2	6.29	1	-4.91	-9.36
Itau CorpBanca ADR	18	0.63	1.56	0.97	1.98	1	-17.21	-23.76
Westpac Banking ADR	14.43	1.35	29.08	0.93	10.32	3	12.94	5.11
Nuveen California Div Advantag Muni	33.85	0.44	0.151	0.02	4.19	5	17.1	20.4
BlackRock Long Term Muni Advantage	15.71	0.83	0.012	0.19	4.64	5	21.56	19.98
Aflac	12.81	4.12	21.94	0.71	2.04	5	15.58	19.14
AG Mortgage Investment	26.24	0.57	0.163	0.95	12.08	1	-5.84	-12.89
AG Mortgage Invest Trust Pb Pref	26.16	0.57	0.163	0.95	12.12	3	4.75	-0.08
AG Mortgage Invest Trust Pa Pref	26.16	0.57	0.163	0.95	12.12	4	5.74	0.83
Federal Agricultural Mortgage	9.37	8.94	0.609	1.19	3.34	5	38.58	20.28
Great Ajax Corp	12.48	1.24	0.118	0.79	8.25	5	29.96	20.83
Alliance Data Systems	7.6	16.12	6.69	1.68	2.06	1	-17.9	-44.78
AllianceBernstein Holding LP	12.02	2.34	0.251	1.19	8.45	3	1.72	-6.96
AllianzGI Diversifiedome Convertibl	9.77	2.32	0.014	1.39	8.84	3	22.72	0.26
AllianzGI Equity Convertible Closed	17.27	1.22	0.018	1.11	7.19	3	14.56	-0.28
Ares Dynamic Credit Allocation Inc	22.69	0.65	0.044	0.55	8.76	1	7.46	-4.08
BlackRock Credit Allocationome Tr	16.86	0.8	0.117	0.34	7.43	5	20.75	13.45
Allstate	14.32	7.55	42.12	0.82	1.85	5	30.09	11.26
Ally Financial Inc	8.19	3.91	5.79	1.29	2.12	5	35.17	19.42
Artisan Partners AM	10.48	2.48	0.792	1.88	9.23	1	21.69	-5.49
New America High Income Closed Fund	10.24	0.89	0.02	0.55	7.28	5	19.18	10.3
Reinsurance of America	13.87	11.3	13.43	0.65	1.79	5	22.44	6.8
Bank of America	10.59	2.81	57.92	1.61	2.42	5	11.75	15.93
Nuveen Build America Bond Closed	30.63	0.71	0.04	0	5.35	5	12.86	12.4
American Financial	13.04	7.86	7.7	0.84	1.76	5	14.59	3.76
First American	12.52	4.76	5.76	0.88	2.82	5	32.82	27.53
...	...	...	...	...	...	...	...	...
Argo Group Int	20.6	3.29	1.91	0.58	1.83	5	1.58	12.93

Table 6.7: Evaluation Table input data for NYSE Financial Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
ABB ADR	43.75	0.41	41.68	1.19	4.33	1	-4.52	-14.63
Accenture	25.05	7.36	43.22	1.05	1.74	5	30.37	15.3
SAP ADR	40.2	2.87	29.3	1.11	1.45	3	15.77	0.63
Infosys ADR	22.1	0.5	12.3	0.48	2.2	5	17.44	14.78
Wipro ADR	16.74	0.22	8.55	0.5	0.29	1	-5.71	-4.22
BT ADR	7.82	1.33	23.46	0.81	9.26	1	-30.46	-31.81
STMicroelectronics ADR	16.03	1.22	9.42	1.41	1.22	5	41.5	17.96
Canon ADR	16.66	1.58	33.8	0.59	5.61	1	-4.71	-14.78
Agilent Technologies	21.76	3.4	5.09	1.45	0.89	5	9.69	11.71
Allegion PLC	23.03	4.37	2.8	1.16	1.07	5	26.33	20.21
Ametek	25.01	3.52	5.04	1.21	0.64	5	29.91	20.81
Amphenol	23.8	4.01	8.33	1.04	1.05	5	17.88	12.33
AO Smith	18.66	2.48	3.08	1.49	1.9	1	8.52	-2.97
Scnc App In	29.82	2.8	5.58	1.31	1.77	5	30.96	17.34
Rockwell Automation	17.01	9.1	6.69	1.42	2.51	2	2.87	-10.41
AVX	9.68	1.6	1.74	1.17	2.98	2	1.38	-6.13
AZZ	18.11	2.13	0.953	1.43	1.76	1	-4.48	-15.61
Badger Meter	41.93	1.24	0.423	0.78	1.31	3	5.87	9.64
Belden	12.48	3.97	2.54	2.37	0.4	1	18.63	-18.62
Regal Beloit	12.62	5.47	3.53	1.6	1.74	1	-1.44	-9.69
Benchmark Electronics	23.01	1.28	2.5	0.85	2.04	5	38.67	26.32
Broadridge	30.08	4.07	4.36	0.7	1.76	5	27.31	4.42
BWX Tech	27.75	1.99	1.79	1.05	1.23	5	44.31	-4.07
CAE Inc.	26.88	0.91	2.58	0.82	1.25	5	32.81	29.22
Jabil Circuit	20.28	1.72	25.28	0.97	0.92	5	40.58	46.86
TE Connectivity	9.37	9.4	13.15	1.17	2.09	4	16.45	11.61
Issuer Direct Corp	61.59	0.17	0.015	0.76	1.95	1	-9.43	-31.92
CTS Corp	20.72	1.51	0.477	1.29	0.51	5	21.05	3.67
Danaher	40.74	3.41	20.25	0.95	0.49	5	34.56	34.22
Deluxe	19.87	2.27	2.01	1.35	2.66	1	17.33	-11.48
DXC Technology	6.7	3.94	20.36	1.96	3.18	1	-49.99	-69.67
Eaton	15.29	5.14	21.71	1.42	3.61	3	14.48	-1.47
Espey Mfg&Electronics	23.47	0.98	0.036	0.23	4.35	1	-7.7	-20.14
Methode Electronics	13.47	2.43	1.05	1.45	1.35	5	40.23	13.13
Emerson	20.23	3.14	18.29	1.37	3.08	3	8.87	-9.55
Energizer	91.94	0.43	2.23	0.67	3.05	1	-12.76	-33.7
Enersys	17.55	3.47	2.92	1.6	1.15	1	-21.48	-23.75
ESCO Technologies	24.93	3.24	0.807	1.07	0.4	5	22.44	32.4
Evertec Inc	23.81	1.31	0.471	0.73	0.64	5	8.4	35.67
FactSet Research	26.55	9.09	1.44	0.95	1.19	4	20.64	13.42
Fortive	36.69	1.77	6.31	1.22	0.43	1	-3.3	-17.39
GlobalSCAPE	21.23	0.55	0.037	0.57	0.52	5	174.9	204.11
Northrop Grumman	18.15	20.26	32.89	0.8	1.44	5	50.57	21.95
Hewlett Packard	19.41	0.73	29.87	1.63	3.19	1	6.81	-8.5
Hexcel	22.98	3.36	2.32	1.04	0.88	5	34.74	26.68
Hill-Rom	32.46	3.02	2.88	0.86	0.86	5	10.71	11.13
HP Inc	5.89	2.72	58.72	1.47	4	1	-21.65	-32.28
Hubbell	19.84	6.63	4.61	1.48	2.56	5	32.37	8.23
IBM	11.75	12.01	77.86	1.36	4.59	4	24.16	0.2
Vishay Intertechnology	8.32	2.09	2.99	1.51	2.18	3	-3.33	-5.12
...	...	...	...	...	...	...	...	...
MSCI	33.64	6.62	1.48	1.12	1.22	5	51.02	44.54

Table 6.8: Evaluation Table input data for NYSE Technology Sector

In tables 6.9 - 6.11, the results of each multicriteria method are presented for each sector. More specifically, the results for the energy sector are presented in 6.9, the results for the technological sector are presented in 6.10 and the results for the financial sector are presented in 6.11.

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Petroleo Brasileiro ADR Reptg 2 Pref	7.69	39.55	14.15	48.63
Phillips 66	45.05	56.91	31.13	57.29
Phillips 66 Partners LP	38.22	51.22	30.92	52.48
Baker Hughes A	-33.91	23.96	-19.06	31.68
GasLog Partners Pref A	50.10	49.62	31.80	50.48
Adams Resources&Energy	-31.25	27.77	-14.89	41.21
Ecopetrol ADR	-10.05	31.04	-5.20	45.89
Total ADR	41.59	40.34	15.27	50.14
Petroleo Brasileiro Petrobras ADR	3.77	36.04	10.80	47.80
CNOOC ADR	46.46	44.38	3.98	54.57
Sinopec Shanghai Petrochemical ADR	19.84	32.62	-6.79	44.53
Royal Dutch Shell ADR	53.86	47.88	18.97	54.97
Equinor ADR	-15.52	33.15	2.12	45.97
ENI ADR	38.02	37.24	15.06	47.96
PetroChina ADR	48.68	43.36	5.50	51.66
Transportadora Gas ADR	40.38	34.41	-6.74	44.76
BP ADR	52.87	43.54	18.17	51.76
Royal Dutch Shell B ADR	53.58	47.59	17.83	54.60
Plains All American Pipeline	-0.31	35.05	5.02	47.65
YPF Sociedad Anonima	-46.86	25.51	-28.10	42.22
Archrock	-56.54	25.91	-7.47	45.01
Teck Resources B	-49.26	26.86	-29.55	43.31
BP Prudhoe Bay Royalty Trust	34.96	37.94	-5.74	44.78
Cabot Oil&Gas	-26.78	30.46	-11.82	43.73
Canadian Natural	-27.97	32.59	-3.75	46.99
Cenovus Energy Inc	-29.00	30.58	-2.69	44.43
Chevron	39.26	45.11	25.05	53.07
Cimarex Energy	-48.09	27.97	-32.74	43.53
CONSOL Coal	33.59	33.40	-2.31	45.33
Concho Resources	-54.79	23.49	-35.52	39.76
ConocoPhillips	-23.01	33.93	-7.10	47.14
Continental Resources	-59.52	23.08	-39.99	40.95
Cosan Ltd	30.66	55.74	22.84	57.70
Crestwood Equity Partners LP	-20.84	38.30	10.80	48.82
Crossamerica Partners LP	32.81	38.39	25.25	47.89
CVR Energy	35.79	49.90	26.43	52.54
Cypress Energy Partners LP	42.26	51.66	32.68	54.35
Delek Logistics Partners LP	28.54	39.40	22.82	48.59
Devon Energy	-64.52	23.77	-32.66	42.93
Encana	-59.82	21.64	-41.06	40.84
Holly Energy Partners LP	19.65	33.10	1.04	44.88
...	...	...	...	...
Williams	-60.82	17.77	-17.75	22.76

Table 6.9: Phase A results for NYSE Energy Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
ABB ADR	11.57	26.99	-22.32	30.96
Accenture	51.66	62.42	38.03	43.95
SAP ADR	1.74	40.47	-3.72	33.49
Infosys ADR	18.75	57.52	23.24	38.55
Wipro ADR	-28.89	23.38	-31.61	29.81
BT ADR	17.89	33.19	-19.79	34.52
STMicroelectronics ADR	10.58	54.31	16.04	39.92
Canon ADR	10.05	32.53	-10.75	34.49
Agilent Technologies	6.24	52.21	7.06	36.48
Allegion PLC	13.61	54.72	18.49	38.85
Ametek	12.99	53.93	16.71	38.90
Amphenol	12.00	54.99	16.37	37.65
AO Smith	-38.05	22.11	-34.04	30.28
Scnc App In	12.27	54.04	16.38	38.58
Rockwell Automation	-14.77	33.92	-13.80	33.17
AVX	-21.38	31.78	-19.73	31.76
AZZ	-41.62	20.76	-40.30	27.80
Badger Meter	-10.87	36.89	-12.37	30.58
Belden	-53.64	17.91	-41.09	29.68
Regal Beloit	-39.90	22.81	-33.96	30.32
Benchmark Electronics	18.31	56.26	24.59	40.38
Broadridge	16.30	56.31	21.43	37.98
BWX Tech	9.47	53.57	14.17	37.62
CAE Inc.	14.35	54.87	21.77	39.49
Jabil Circuit	25.64	58.72	33.93	43.90
TE Connectivity	19.82	52.45	10.45	39.97
Issuer Direct Corp	-42.43	18.35	-44.52	21.92
CTS Corp	4.02	51.20	5.86	35.71
Danaher	23.20	56.17	29.11	40.79
Deluxe	-35.54	23.27	-30.23	30.79
DXC Technology	-46.48	20.48	-38.87	25.97
...	...	...	...	...
Xerox	-3.92	48.70	4.03	40.19

Table 6.10: Phase A results for NYSE Technology Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Nuveen CA MVF 2	38.46	46.75	1.96	28.30
Nuveen High Income 2020 Target Term	25.03	39.12	-7.67	29.03
Nuveen Dow 30Sm	5.26	39.64	-7.08	28.91
Ellsworth Growth Pref A	170.49	47.55	3.71	30.42
Federal Agricultural Mortgage A	117.01	48.83	16.19	33.06
Chimera Investment Pref A	-301.05	48.96	13.82	29.95
Ares Management Pref A	-100.73	40.85	-12.01	27.03
Apollo Global Management A	-206.36	49.72	14.39	35.24
Ladder Capital A	95.02	46.88	5.62	29.40
Aberdeen Emerging Markets Equity	125.01	41.73	-5.66	30.79
Aberdeen Asia-Pacific	-20.04	37.36	-8.76	26.31
Adams Diversified Equity Closed	163.72	46.79	0.19	31.29
Barclays ADR	-28.00	32.83	-20.10	28.61
Santander Chile ADR	-148.79	33.15	-17.59	27.93
Sumitomo Mitsui Financial ADR	62.70	40.39	-6.28	29.70
Mitsubishi UFJ Financial ADR	-42.05	33.11	-18.40	30.64
China Life Insurance ADR	-10.27	29.70	-32.27	32.63
Aegon ADR	-57.21	37.19	-0.50	33.61
Banco Bilbao ADR	74.45	33.46	-17.61	29.43
Credit Suisse ADR	-133.90	30.77	-25.57	29.27
Prudential Public ADR	-240.19	20.80	-35.21	23.81
Lloyds Banking ADR	-44.85	35.85	-5.50	32.11
ING ADR	-51.04	33.20	-15.32	30.23
BBVA Banco Frances ADR	361.38	41.06	0.82	31.81
Santander ADR	-64.06	34.57	-12.04	32.36
Itau CorpBanca ADR	-154.06	26.98	-38.60	24.33
Westpac Banking ADR	66.33	44.23	8.37	32.13
Nuveen California Div Advantag Muni	-133.58	50.80	14.67	30.98
BlackRock Long Term Muni Advantage	128.35	52.47	19.33	32.74
Aflac	109.97	50.37	16.05	33.31
AG Mortgage Investment	-242.42	32.11	-16.78	26.50
AG Mortgage Invest Trust Pb Pref	-326.60	40.67	-1.94	28.96
AG Mortgage Invest Trust Pa Pref	-364.20	43.98	1.55	29.05
Federal Agricultural Mortgage	131.64	50.89	23.55	34.86
Great Ajax Corp	140.96	52.22	23.22	33.46
Alliance Data Systems	46.65	25.47	-33.43	27.20
AllianceBernstein Holding LP	5.02	38.93	-8.49	28.66
AllianzGI Diversifiedome Convertibl	38.71	41.01	-1.25	30.88
AllianzGI Equity Convertible Closed	-52.78	39.95	-5.22	29.30
Ares Dynamic Credit Allocation Inc	-324.81	35.28	-5.84	28.43
BlackRock Credit Allocationome Tr	86.51	51.94	21.29	31.89
...	...	...	...	...
Bancroft	157.38	49.87	12.73	32.49

Table 6.11: Phase A results for NYSE Financial Sector

Given the results of the multicriteria methods, the final step of the first phase involves the cumulative ranking of the securities. The cumulative ranking is graphically presented in figures 6.1 - 6.4.

Figure 6.1 shows the cumulative ranking for the securities that belong to the technological sector.

Figure 6.2 shows the cumulative ranking for the securities that belong to the energy sector.

Finally, figure 6.3 shows the cumulative ranking for the securities that belong to the financial sector. While in the first two cases the number of securities is not too large, the financial sector contains a large amount of securities thus making the visualisation indistinct. Therefore, figure 6.4, which includes only the most favourable securities of the financial sector, has been incorporated.

Finally, the first phase terminates with the formulation of the most suitable portfolio for each sector. Making the assumption that we select the 20-highest ranked securities from each industrial sector, the portfolio has been formulated as shown in table 6.12.

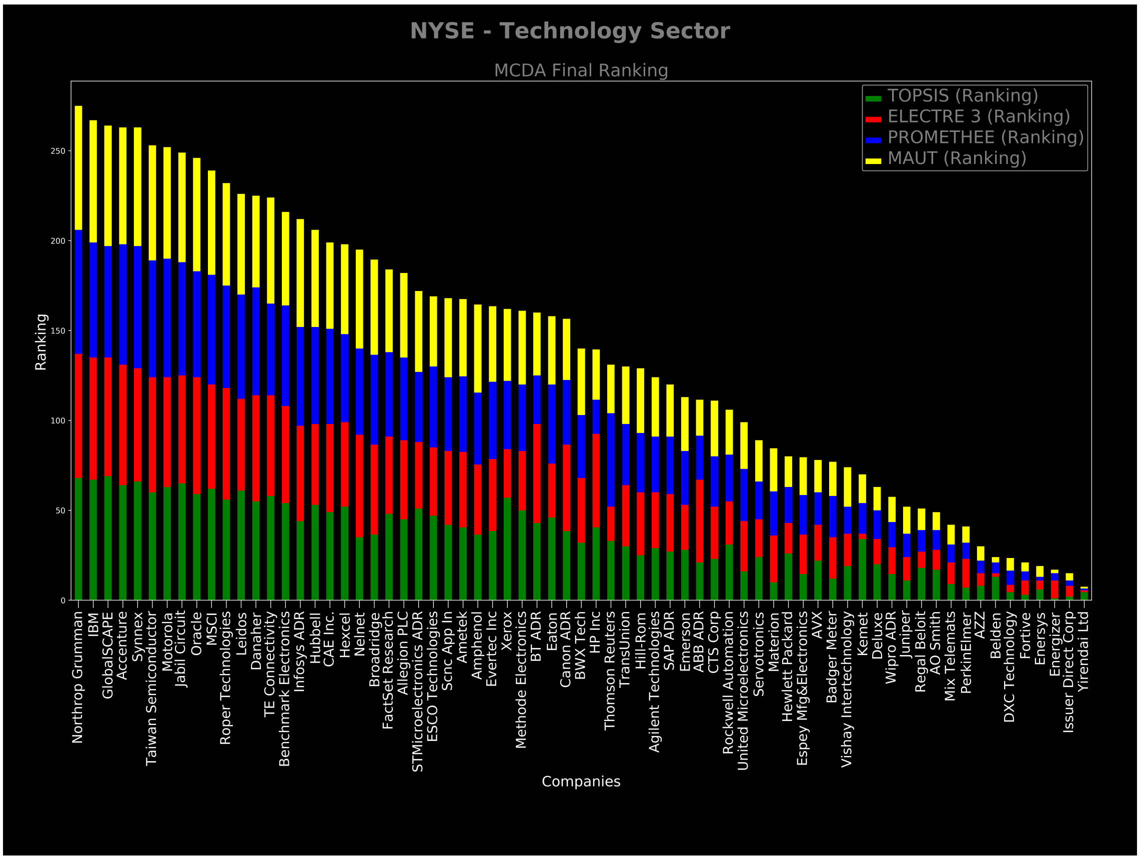


Figure 6.1: Cumulative ranking for NYSE technology securities

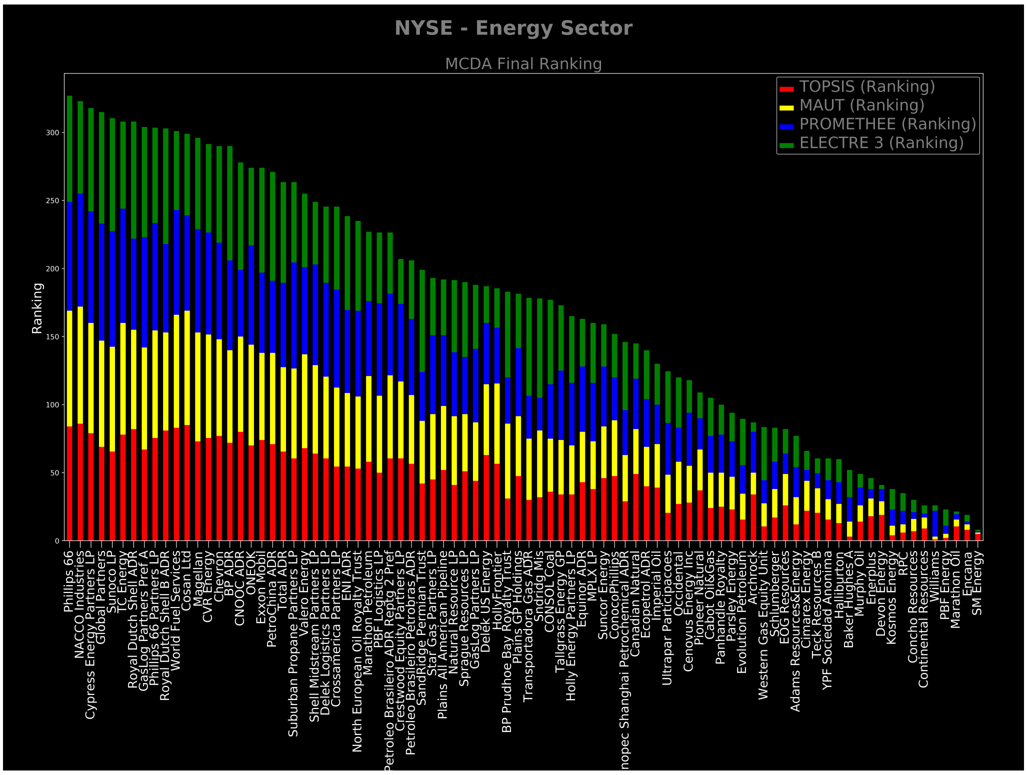


Figure 6.2: Cumulative ranking for NYSE energy securities

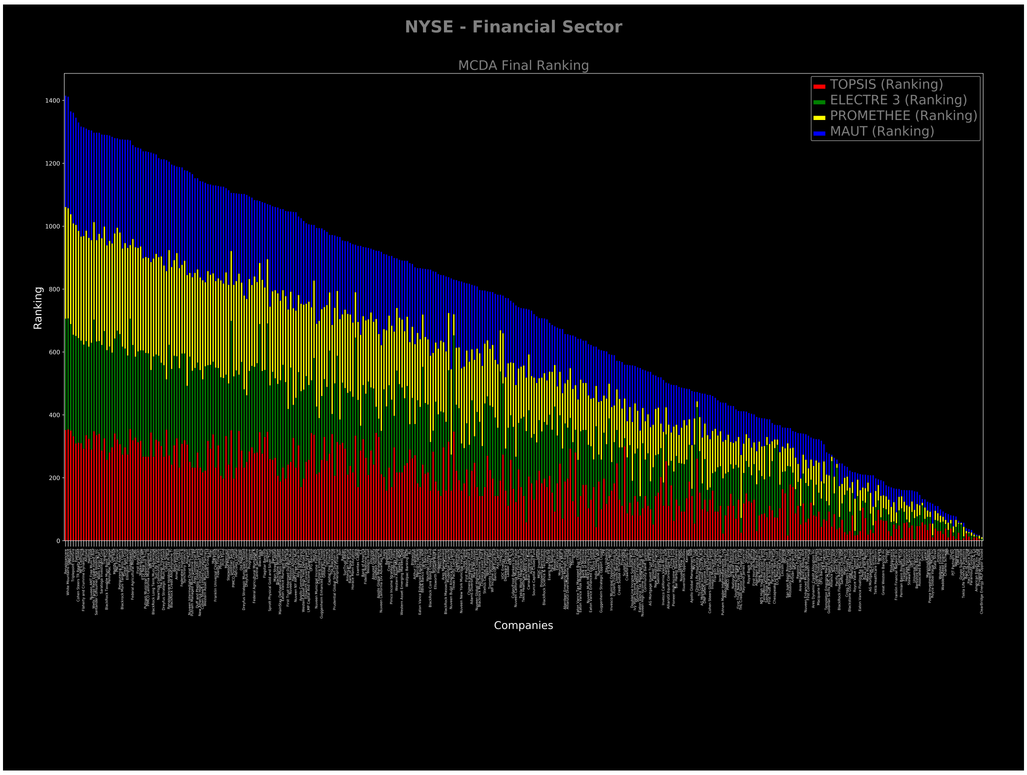


Figure 6.3: Cumulative ranking for NYSE financial securities

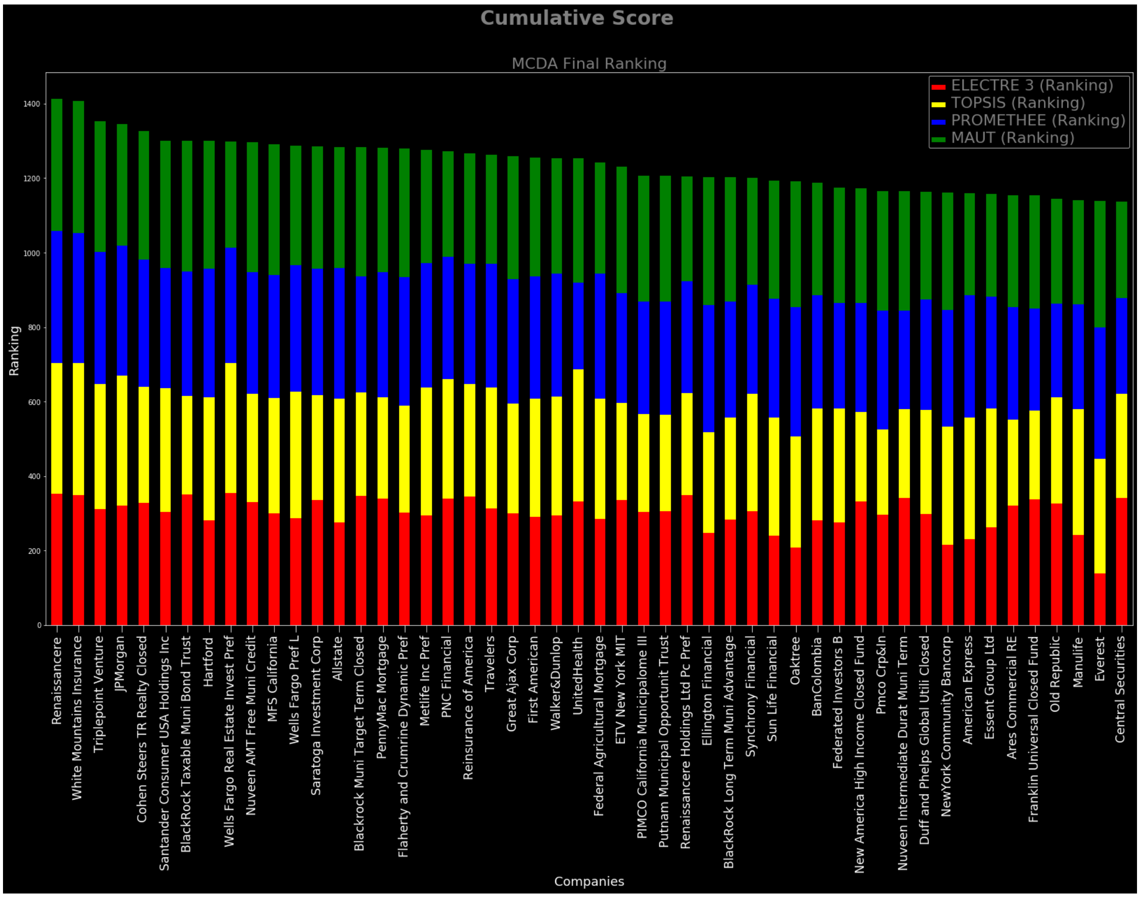


Figure 6.4: Cumulative ranking for the most dominant NYSE energy securities

Technology	**Energy**	**Financial**
Northrop Grumman	Phillips 66	Renaissancere
GlobalSCAPE	NACCO Industries	White Mountains Insurance
Accenture	Cypress Energy Partners LP	Triplepoint Venture
Synnex	Global Partners	JPMorgan
IBM	Sunoco LP	Cohen Steers TR Realty Closed
Taiwan Semiconductor	TC Energy	Santander Consumer USA Holdings Inc
Motorola	Royal Dutch Shell ADR	BlackRock Taxable Muni Bond Trust
Jabil Circuit	GasLog Partners Pref A	Hartford
Oracle	Phillips 66 Partners LP	Wells Fargo Real Estate Invest Pref
MSCI	Royal Dutch Shell B ADR	Nuveen AMT Free Muni Credit
Roper Technologies	World Fuel Services	MFS California
Danaher	Cosan Ltd	Wells Fargo Pref L
Leidos	Magellan	Saratoga Investment Corp
Benchmark Electronics	CVR Energy	Allstate
Infosys ADR	BP ADR	Blackrock Muni Target Term Closed
Hubbell	Chevron	PennyMac Mortgage
Nelnet	CNOOC ADR	Flaherty and Crumrine Dynamic Pref
CAE Inc.	Exxon Mobil	Metlife Inc Pref
Hexcel	ONEOK	PNC Financial
Broadridge	PetroChina ADR	Reinsurance of America

Table 6.12: Selected securities from NYSE stock exchange

Figures 6.5 - 6.7 depict the security values for all companies that participate in the process during the time horizon of the experiment. The visualisation of the security values is an important step of the process, as it demonstrates the characteristics of the securities. Table 6.13 includes the most significant financial indexes for the selected securities.

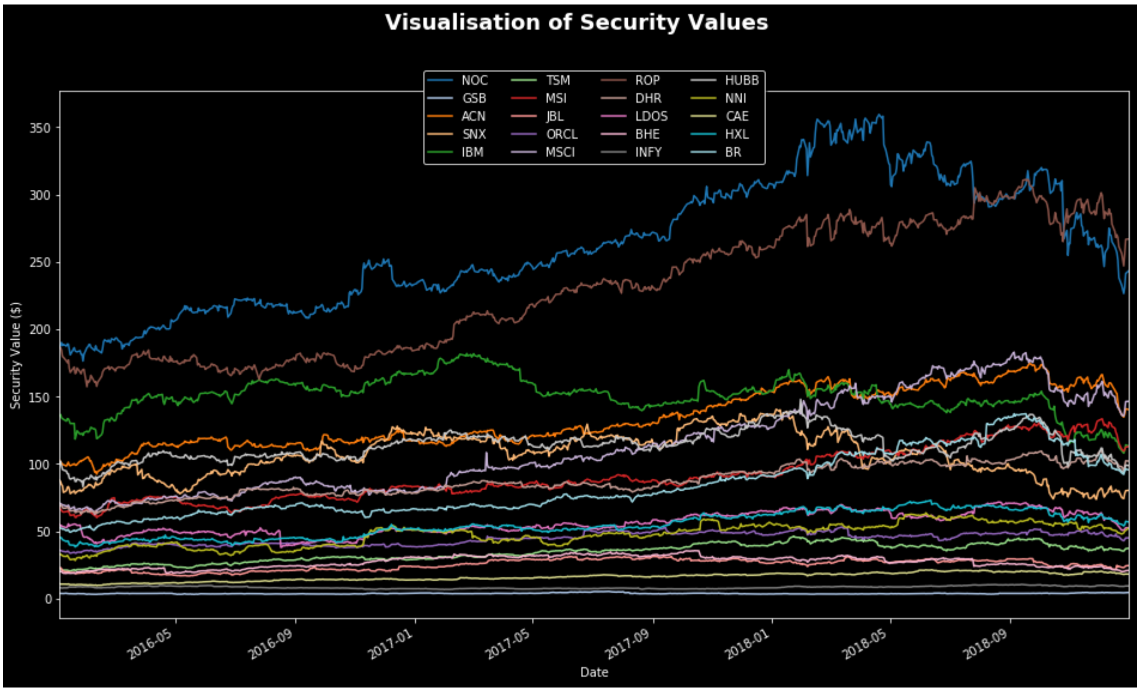


Figure 6.5: Value Visualisation for NYSE technology securities

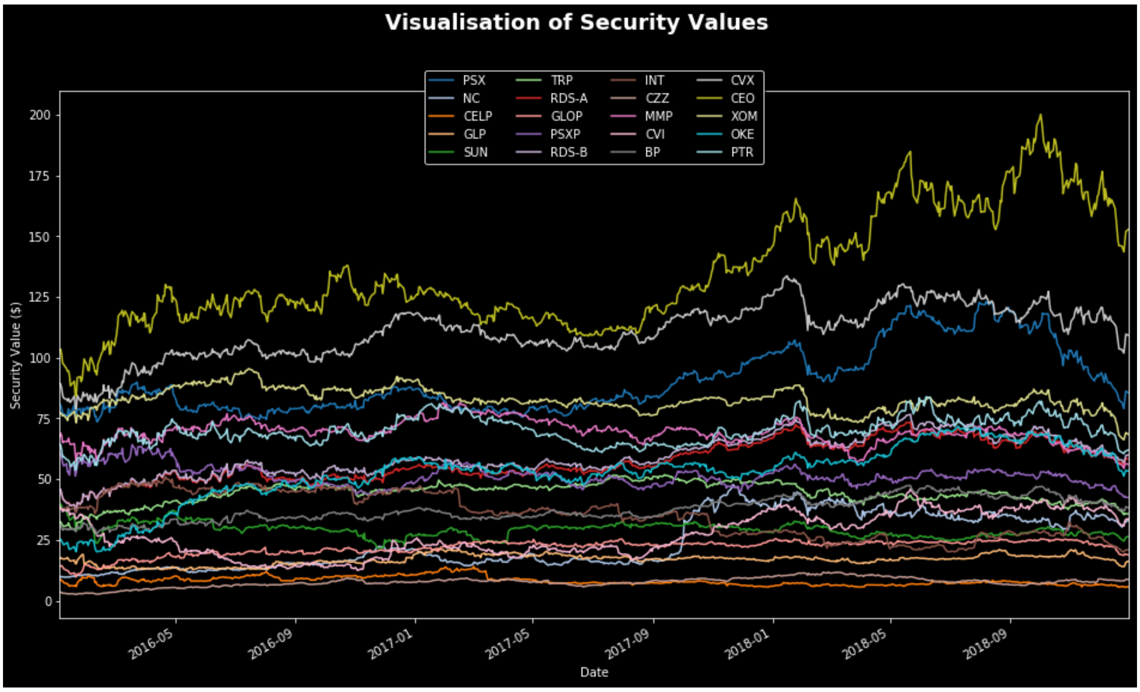


Figure 6.6: Value Visualisation for NYSE energy securities

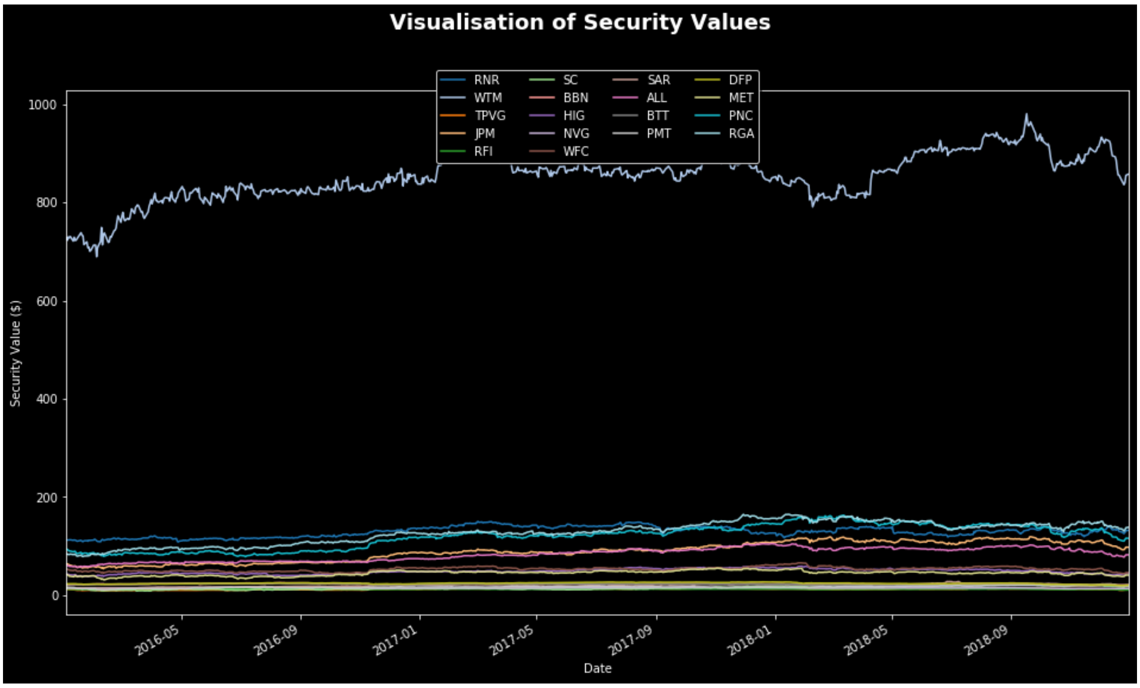


Figure 6.7: Value Visualisation for NYSE financial securities

	MinRet	MaxRet	Median	Mean	SD	VaR99	VaR97	VaR95	Skewness	Kurtosis	MinPerSD
NOC	-0.077808	0.063431	0.000676	0.000448	0.013467	-0.049242	-0.024005	-0.019069	-0.674012	6.010290	5.777613
GSB	-0.145414	0.112329	0.000000	0.000529	0.025610	-0.068713	-0.049494	-0.037134	-0.178767	3.616340	5.678001
ACN	-0.056337	0.051843	0.001509	0.000480	0.011213	-0.035905	-0.022927	-0.017429	-0.658652	3.198245	5.024272
SNX	-0.106592	0.083184	0.001181	0.000057	0.018689	-0.062642	-0.037670	-0.027143	-0.647182	4.394566	5.703297
IBM	-0.089540	0.071541	0.000000	-0.000162	0.012330	-0.036385	-0.022937	-0.017667	-0.478771	7.086411	7.261800
TSM	-0.073449	0.052043	0.000766	0.000795	0.014376	-0.039892	-0.026994	-0.022551	-0.310216	1.943694	5.109037
MSI	-0.054133	0.058159	0.001211	0.000757	0.012276	-0.035977	-0.026263	-0.018172	-0.169242	2.672691	4.409556
JBL	-0.083417	0.134342	0.001056	0.000260	0.017974	-0.050843	-0.036011	-0.029023	0.238747	7.066522	4.640916
ORCL	-0.107381	0.084395	0.000842	0.000388	0.013077	-0.039936	-0.027570	-0.019280	-0.578328	11.245113	8.211162
MSCI	-0.084600	0.142090	0.001597	0.001079	0.015456	-0.039073	-0.028230	-0.021382	0.579821	12.724204	5.473484
ROP	-0.083342	0.051727	0.000913	0.000553	0.012257	-0.034488	-0.022477	-0.018602	-0.770306	6.374014	6.799291
DHR	-0.060621	0.063723	0.000638	0.000576	0.010544	-0.029102	-0.019503	-0.015026	0.017092	5.072333	5.749169
LDOS	-0.234929	0.080472	0.000780	0.000090	0.017017	-0.050421	-0.026742	-0.022870	-3.663357	49.360685	13.805557
BHE	-0.153169	0.056000	0.001546	0.000182	0.016489	-0.050639	-0.031140	-0.024771	-1.902171	13.712781	9.289050
INFY	-0.079694	0.066125	0.000000	0.000287	0.014153	-0.036931	-0.027278	-0.021654	-0.212132	3.641736	5.631038
HUBB	-0.119682	0.069599	0.000000	0.000088	0.014425	-0.035496	-0.026763	-0.021403	-0.945870	10.359222	8.296801
NNI	-0.097416	0.127130	0.001553	0.000784	0.018295	-0.048459	-0.032050	-0.026833	0.186857	6.357661	5.324801
CAE	-0.052734	0.057234	0.000978	0.000734	0.012595	-0.034766	-0.023228	-0.019422	-0.077775	2.497146	4.186947
HXL	-0.063379	0.080656	0.000662	0.000406	0.014856	-0.038635	-0.027071	-0.022961	0.187992	3.901582	4.266157
BR	-0.100575	0.089993	0.000853	0.000866	0.012592	-0.032041	-0.021722	-0.016875	-0.418587	10.824529	7.987370
PSX	-0.052226	0.049867	0.000263	0.000156	0.013342	-0.035732	-0.029424	-0.022674	-0.069855	1.821956	3.914278
NC	-0.088623	0.282276	0.002196	0.002056	0.028773	-0.058687	-0.050514	-0.040731	2.434129	22.467297	3.080038
CELP	-0.207407	0.252144	0.000000	0.000172	0.038552	-0.091408	-0.071286	-0.060333	0.572526	6.099420	5.379910
GLP	-0.206842	0.106250	0.000000	0.000207	0.023876	-0.053926	-0.039066	-0.034394	-0.404502	8.979767	8.663231
SUN	-0.124734	0.152829	0.000000	-0.000267	0.021671	-0.054210	-0.036399	-0.030947	0.492752	9.187671	5.755837
TRP	-0.044504	0.073667	0.000583	0.000225	0.012816	-0.031137	-0.024725	-0.020866	0.150190	1.782739	3.472476
RDS-A	-0.079804	0.097415	0.000779	0.000454	0.015527	-0.037409	-0.028919	-0.024275	0.232366	4.328223	5.139826
GLOP	-0.124204	0.118321	0.000432	0.000592	0.020525	-0.053575	-0.035926	-0.029338	0.029716	5.040865	6.051268
PSXP	-0.076577	0.072812	0.000227	-0.000340	0.016672	-0.041726	-0.030632	-0.027324	0.064466	1.811138	4.593196
RDS-B	-0.088053	0.102781	0.001063	0.000488	0.015950	-0.038365	-0.028660	-0.025406	0.231354	5.102997	5.520447
INT	-0.191685	0.155319	0.000776	-0.000544	0.022117	-0.064801	-0.039957	-0.029993	-0.994850	16.387535	8.666700
CZZ	-0.149343	0.129114	0.001208	0.001521	0.025560	-0.066787	-0.042760	-0.034030	0.060838	3.691457	5.842848
MMP	-0.070588	0.062347	-0.000134	-0.000142	0.014212	-0.036271	-0.026411	-0.023050	0.204543	2.775524	4.966742
CVI	-0.129289	0.217920	0.000450	0.000220	0.028745	-0.070141	-0.052375	-0.045590	0.747455	6.694582	4.497853
BP	-0.078008	0.073658	0.000286	0.000404	0.015362	-0.040129	-0.028719	-0.024827	-0.112656	2.572632	5.077860
CVX	-0.057246	0.071827	0.000537	0.000348	0.013054	-0.032956	-0.024475	-0.021098	-0.029018	2.861619	4.385386
CEO	-0.066644	0.089765	0.000671	0.000682	0.017798	-0.047548	-0.033109	-0.027702	0.018867	1.796554	3.744422
XOM	-0.057277	0.044560	0.000357	-0.000107	0.010948	-0.032464	-0.022010	-0.018559	-0.359791	2.129880	5.231755
OKE	-0.102469	0.173115	0.001211	0.001221	0.020016	-0.051275	-0.034962	-0.029202	0.618880	9.150568	5.119342
PTR	-0.060679	0.079934	-0.000577	0.000088	0.016750	-0.042951	-0.030406	-0.025048	0.368817	2.304897	3.622653
RNR	-0.081077	0.074774	0.000495	0.000293	0.012492	-0.035414	-0.022093	-0.017910	-0.126635	6.468945	6.490299
WTM	-0.047301	0.048832	0.000000	0.000263	0.009417	-0.022648	-0.016780	-0.014517	0.239646	3.256256	5.022833
TPVG	-0.073724	0.075510	0.000000	0.000016	0.014773	-0.042986	-0.028417	-0.023696	-0.223974	3.070499	4.990380
JPM	-0.059050	0.046361	0.000816	0.000650	0.013320	-0.037998	-0.025805	-0.020167	-0.246559	2.147910	4.433260
RFI	-0.038693	0.034331	0.000000	-0.000151	0.008889	-0.026188	-0.017003	-0.015284	-0.351757	1.720516	4.352864
SC	-0.176916	0.159692	0.000000	0.000474	0.025096	-0.071158	-0.043279	-0.034471	-0.018554	9.003217	7.049472
BBN	-0.036272	0.042471	0.000437	0.000045	0.007102	-0.022376	-0.015489	-0.010964	-0.250472	4.240600	5.107474
HIG	-0.065277	0.056965	0.000245	0.000123	0.012738	-0.035785	-0.025203	-0.021255	-0.105075	3.405590	5.124597
NVG	-0.030447	0.025381	0.000000	-0.000043	0.005344	-0.014984	-0.011292	-0.008829	-0.469559	3.911184	5.696834
WFC	-0.101485	0.056119	0.000000	-0.000097	0.013671	-0.039483	-0.024013	-0.021315	-0.530029	5.006321	7.423595
SAR	-0.084481	0.069110	0.000411	0.000484	0.015293	-0.040435	-0.030324	-0.023445	-0.231054	3.138825	5.524081
ALL	-0.054447	0.061405	0.000770	0.000454	0.010449	-0.030831	-0.020749	-0.015893	-0.258645	4.973240	5.210772
BTT	-0.036693	0.032045	0.000000	-0.000063	0.005571	-0.014312	-0.010154	-0.008074	-0.183475	6.377144	6.586755
PMT	-0.120473	0.061785	0.000973	0.000395	0.013679	-0.040792	-0.027168	-0.022181	-1.322685	10.234292	8.806875
DFP	-0.034714	0.035406	0.000372	-0.000106	0.007421	-0.021357	-0.016318	-0.012610	-0.350134	2.451549	4.677797
MET	-0.080176	0.085203	0.000435	0.000093	0.016102	-0.050673	-0.034506	-0.026505	-0.078583	4.120069	4.979360
PNC	-0.059738	0.045776	0.000617	0.000367	0.013296	-0.041085	-0.026321	-0.021236	-0.485273	2.093130	4.492779
RGA	-0.056277	0.053854	0.000842	0.000743	0.012492	-0.037142	-0.024446	-0.019967	-0.196426	2.613224	4.505104

Table 6.13: Statistical indexes for the selected securities

6.3.2 Phase II: Multiobjective portfolio optimisation

The second phase of the methodological framework includes the portfolio optimisation process. Therefore, the objective of this phase is the determination of the proportion of each security in the portfolio, given a set of securities which were selected in phase I. The results of the first phase (identification of the securities which serve as the best investment prospects) are the input for the second phase of the proposed methodology.

The portfolio includes securities from all three industrial sectors. Therefore, the number of securities that will participate in the second phase of the process is 60. More specifically, we select the 20 most favourable companies from each industrial sector. Phase II includes four different methods for portfolio optimisation: (a) mean - variance MIQP model, (b) goal programming model, (c) PROMETHEE flow multiobjective model and (d) genetic algorithm model.

Method 1: Mean - Variance MIQP model

The first method is based on a variation of the mean - variance approach, which is extended with additional constraints. More specifically, the imposed constraints are the following:

1. Minimum number of securities to participate in a portfolio equal to 4.
2. Maximum number of securities to participate in a portfolio equal to 40 of the total number of securities.
3. Minimum percentage of capital invested in a security (if this security participates to the portfolio) equal to 0.05%.
4. Maximum percentage of capital invested in a security (if this security participates to the portfolio) equal to 40%.
5. Minimum percentage of capital invested on a specific industrial sector equal to 5%.
6. Maximum percentage of capital invested on a specific industrial sector equal to 70%.

Figure 6.8 shows a 3-dimensional visualisation of the efficient portfolios. More specifically, x-axis represents the pareto optimal portfolios, y-axis represents the 20 securities, while z-axis demonstrates the percentage of capital investment.

The minimum volatility and maximum sharpe ratio portfolios are depicted in figure 6.9.

Table 6.14 shows the composition of each pareto optimal portfolio, i.e. the percentage of capital invested in each security.

Finally, figure 6.10 shows the most dominant securities. The horizontal axis denotes the average participation of each security in the portfolios. Therefore, the securities which are placed in the right part of the figure participate in the majority of the pareto optimal portfolio. The vertical axis denotes the average proportion of each security in the portfolios. The securities which are placed in the upper section of the figure participate with a greater proportion in the portfolios than the securities which are placed lower. Conclusively, the securities which are placed in the upper right section of the figure are the best investment options.

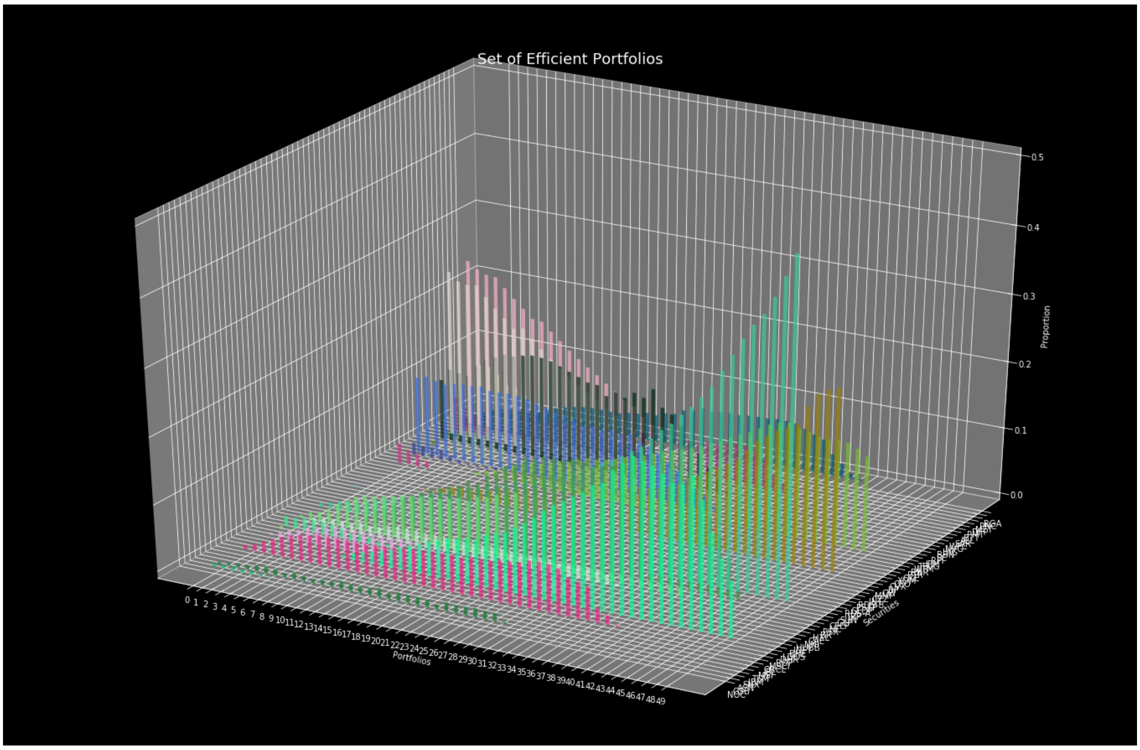
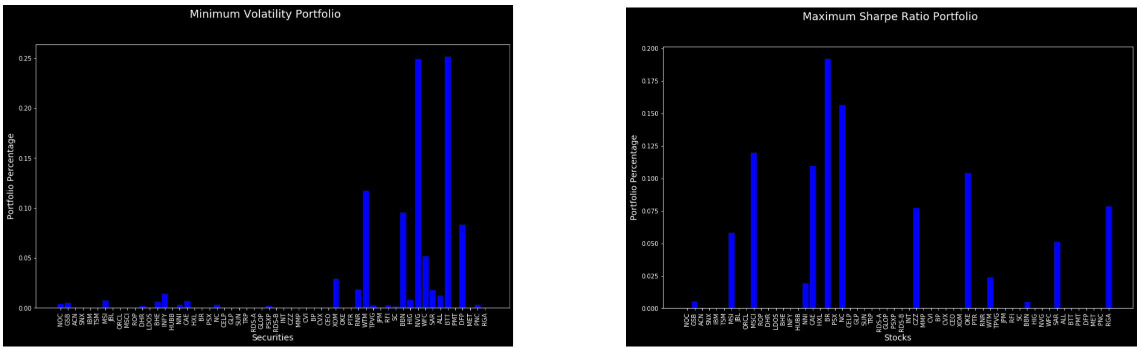


Figure 6.8: Visualisation of efficient portfolios for NYSE stock exchange



(a) Minimum volatility portfolio for NYSE stock exchange

(b) Maximum sharpe ratio portfolio for NYSE stock exchange

Figure 6.9: GMVP & Max Sharpe Ratio portfolios

Portf	NOC	GSB	ACN	SNX	IBM	TSM	MSI	JBL	ORCL	MSCI	ROP	DHR	LDOS	BHE	...	DFP	MET	PNC	RGA
1	0.01	0.00	0.0	0.0	0.0	0.0	0.01	0.0	0.0	0.00	0.0	0.00	0.0	0.01	...	0.08	0.0	0.00	0.00
2	0.01	0.00	0.0	0.0	0.0	0.0	0.01	0.0	0.0	0.00	0.0	0.01	0.0	0.00	...	0.08	0.0	0.01	0.00
3	0.01	0.01	0.0	0.0	0.0	0.0	0.02	0.0	0.0	0.00	0.0	0.01	0.0	0.00	...	0.07	0.0	0.01	0.00
4	0.00	0.01	0.0	0.0	0.0	0.0	0.02	0.0	0.0	0.00	0.0	0.02	0.0	0.00	...	0.06	0.0	0.00	0.00
5	0.01	0.01	0.0	0.0	0.0	0.0	0.03	0.0	0.0	0.00	0.0	0.02	0.0	0.00	...	0.06	0.0	0.00	0.00
6	0.00	0.01	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.00	0.0	0.03	0.0	0.00	...	0.05	0.0	0.00	0.01
7	0.00	0.01	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.00	0.0	0.03	0.0	0.00	...	0.04	0.0	0.00	0.02
8	0.00	0.01	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.00	0.0	0.03	0.0	0.00	...	0.03	0.0	0.00	0.02
9	0.00	0.01	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.00	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
10	0.00	0.01	0.0	0.0	0.0	0.0	0.04	0.0	0.0	0.00	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
11	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.01	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
12	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.01	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
13	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.02	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
14	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.02	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
15	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.03	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.03
16	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.03	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.04
17	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.04	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.04
18	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.05	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.04
19	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.05	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.04
20	0.00	0.01	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.06	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.05
21	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.06	0.0	0.03	0.0	0.00	...	0.00	0.0	0.00	0.05
22	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.06	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.06
23	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.07	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.06
24	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.07	0.0	0.01	0.0	0.00	...	0.00	0.0	0.00	0.07
25	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.08	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.06
26	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.08	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.06
27	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.09	0.0	0.01	0.0	0.00	...	0.00	0.0	0.00	0.07
28	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.10	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.07
29	0.00	0.01	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.11	0.0	0.01	0.0	0.00	...	0.00	0.0	0.00	0.07
30	0.00	0.01	0.0	0.0	0.0	0.0	0.07	0.0	0.0	0.12	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.07
31	0.00	0.00	0.0	0.0	0.0	0.0	0.07	0.0	0.0	0.13	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.07
32	0.00	0.00	0.0	0.0	0.0	0.0	0.07	0.0	0.0	0.14	0.0	0.02	0.0	0.00	...	0.00	0.0	0.00	0.07
33	0.00	0.00	0.0	0.0	0.0	0.0	0.07	0.0	0.0	0.15	0.0	0.01	0.0	0.00	...	0.00	0.0	0.00	0.07
34	0.00	0.00	0.0	0.0	0.0	0.0	0.07	0.0	0.0	0.16	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.07
35	0.00	0.00	0.0	0.0	0.0	0.0	0.06	0.0	0.0	0.17	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.06
36	0.00	0.00	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.18	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.06
37	0.00	0.00	0.0	0.0	0.0	0.0	0.05	0.0	0.0	0.19	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.05
38	0.00	0.00	0.0	0.0	0.0	0.0	0.02	0.0	0.0	0.21	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.03
39	0.00	0.00	0.0	0.0	0.0	0.0	0.01	0.0	0.0	0.22	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.02
40	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.24	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.01
41	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.23	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
42	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.23	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
43	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.22	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
44	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.21	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
45	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.21	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
46	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.23	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
47	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.19	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
48	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.15	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
49	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.11	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00
50	0.00	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.0	0.08	0.0	0.00	0.0	0.00	...	0.00	0.0	0.00	0.00

Table 6.14: Set of efficient portfolios for NYSE stock exchange

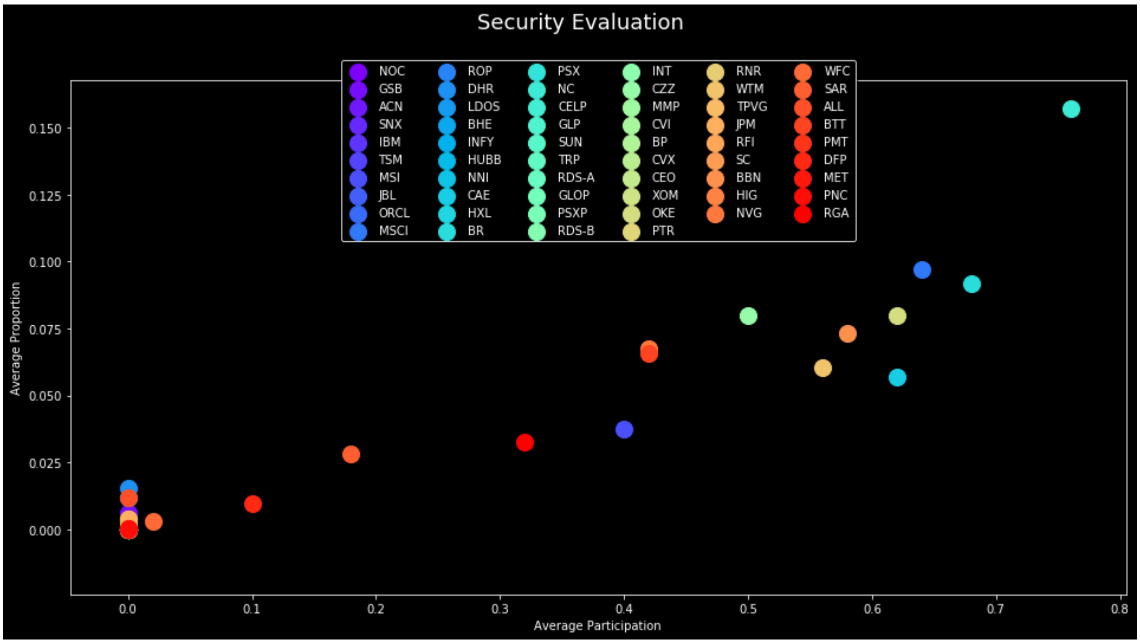


Figure 6.10: NYSE securities comparative evaluation

Method 2: Goal programming model

The second method is a goal programming model, with the following constraints:

1. Goal 1: Portfolio beta set equal to 0.9
2. Goal 2: Portfolio dividend yield set equal to 1.5 %
3. Goal 3: Percentage of securities with revenue $\geq$ 30 billions set equal to 50 %.
4. Minimum percentage of capital invested in a security (if this security participates to the portfolio) equal to 0.03 %.
5. Maximum percentage of capital invested in a security (if this security participates to the portfolio) equal to 20 %.
6. Minimum number of securities to participate in a portfolio equal to 20.
7. Maximum number of securities to participate in a portfolio equal to 40.

The three goals were equipped with deviational variables, while the four constraints are strict. The goal programming model resulted in the portfolio presented in table 6.15.

NOC	GSB	ACN	SNX	IBM	TSM	MSI	JBL	ORCL	MSCI
0.1158	0.0300	0.0	0.0	0.0	0.0	0.0300	0.0	0.2000	0.0
ROP	DHR	LDOS	BHE	INFY	HUBB	NNI	CAE	HXL	BR
0.0300	0.0299	0.0	0.0443	0.0356	0.0	0.0300	0.0300	0.0	0.0305
PSX	NC	CELP	GLP	SUN	TRP	RDS-A	GLOP	PSXP	RDS-B
0.0	0.0300	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
INT	CZZ	MMP	CVI	BP	CVX	CEO	XOM	OKE	PTR
0.0300	0.0300	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
RNR	WTM	TPVG	JPM	RFI	SC	BBN	HIG	NVG	WFC
0.0300	0.0300	0.0	0.0	0.0	0.0	0.0	0.0300	0.0	0.0
SAR	ALL	BTT	PMT	DFP	MET	PNC	RGA		
0.0	0.1541	0.0300	0.0	0.0	0.0	0.0	0.0300		

Table 6.15: Goal programming portfolio for NYSE stock exchange

Method 3: PROMETHEE flow multiobjective programming model

The third method is a biobjective programming model which includes two objective functions: (a) the PROMETHEE net flow of the alternatives and (b) the portfolio beta. The model is equipped with the following constraints:

1. Minimum percentage of capital invested in a security (if this security participates to the portfolio) equal to 0.03 %.

2. Maximum percentage of capital invested in a security (if this security participates to the portfolio) equal to 20 %.

3. Minimum number of securities to participate in a portfolio equal to 20.

4. Maximum number of securities to participate in a portfolio equal to 40.

The problem was solved parametrically, setting portfolio beta as a parameter. The efficient frontier is presented in the following figure.

The efficient portfolios are presented in the following table.

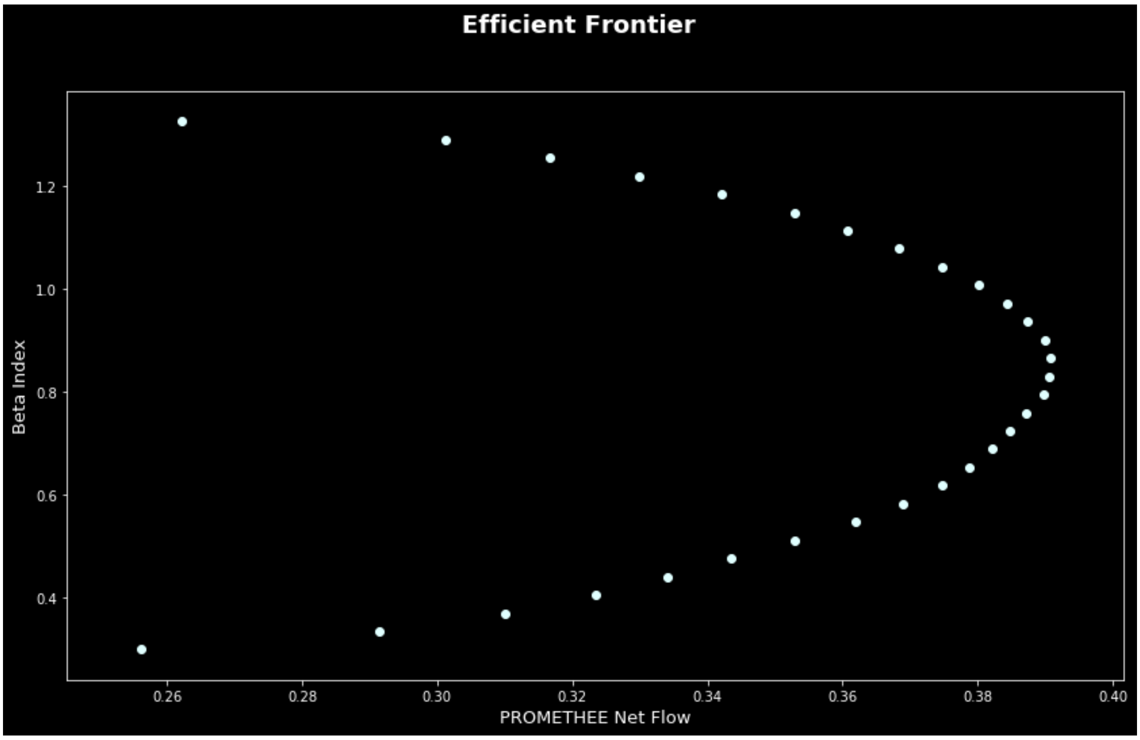


Figure 6.11: Visualisation of efficient portfolios for NYSE stock exchange

Method 4: Genetic algorithm model

Finally, the fourth method is a genetic algorithm model. This method differs from the previous methods, as it is a passive strategy for portfolio optimisation. More specifically, the returns of the securities are compared to the returns of the market index. The target is to maximise the number of time periods that the constructed portfolio beats the market index.

The resulting portfolio of this method is presented in the following table.

Portf	NOC	GSB	ACN	SNX	IBM	TSM	MSI	JBL	ORCL	MSCI	ROP	DHR	LDOS	BHE	...	DFP	MET	PNC	RGA
1	0.00	0.03	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.03
2	0.03	0.03	0.00	0.03	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
3	0.03	0.03	0.00	0.03	0.00	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
4	0.03	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
5	0.09	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
6	0.14	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
7	0.18	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
8	0.20	0.03	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
9	0.20	0.03	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
10	0.20	0.03	0.03	0.08	0.00	0.03	0.03	0.03	0.00	0.00	0.00	0.00	0.00	0.0	...	0.03	0.00	0.00	0.00
11	0.20	0.03	0.03	0.10	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.0	...	0.00	0.00	0.00	0.00
12	0.20	0.03	0.03	0.11	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.0	...	0.00	0.00	0.00	0.00
13	0.20	0.03	0.03	0.16	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.0	...	0.00	0.00	0.00	0.00
14	0.20	0.03	0.03	0.17	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.0	...	0.00	0.00	0.00	0.00
15	0.20	0.03	0.03	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.0	...	0.00	0.00	0.00	0.00
16	0.20	0.03	0.04	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.0	...	0.00	0.00	0.00	0.00
17	0.20	0.03	0.04	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.0	...	0.00	0.00	0.00	0.00
18	0.20	0.03	0.09	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.0	...	0.00	0.00	0.00	0.00
19	0.20	0.03	0.09	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.03	0.0	...	0.00	0.00	0.00	0.00
20	0.20	0.00	0.08	0.20	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
21	0.20	0.00	0.04	0.20	0.03	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
22	0.20	0.00	0.04	0.20	0.03	0.03	0.03	0.03	0.00	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
23	0.18	0.00	0.03	0.20	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
24	0.13	0.00	0.03	0.20	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
25	0.09	0.00	0.03	0.20	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.03	0.0	...	0.00	0.00	0.00	0.00
26	0.04	0.00	0.03	0.20	0.03	0.03	0.00	0.03	0.03	0.03	0.00	0.00	0.08	0.0	...	0.00	0.00	0.00	0.00
27	0.03	0.00	0.03	0.17	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.12	0.0	...	0.00	0.00	0.00	0.00
28	0.03	0.00	0.03	0.10	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.19	0.0	...	0.00	0.00	0.00	0.00
29	0.03	0.00	0.03	0.04	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.20	0.0	...	0.00	0.00	0.00	0.00
30	0.00	0.00	0.03	0.00	0.03	0.00	0.00	0.00	0.03	0.03	0.03	0.00	0.20	0.0	...	0.00	0.03	0.03	0.00

Table 6.16: Set of efficient portfolios for NYSE stock exchange with MOIP PROMETHEE method

NOC	GSB	ACN	SNX	IBM	TSM	MSI	JBL	ORCL	MSCI
0.1	0.005	0.003	0.209	0.004	0.0	0.003	0.005	0.003	0.004
ROP	DHR	LDOS	BHE	INFY	HUBB	NNI	CAE	HXL	BR
0.322	0.003	0.003	0.005	0.004	0.004	0.003	0.004	0.196	0.003
PSX	NC	CELP	GLP	SUN	TRP	RDS-A	GLOP	PSXP	RDS-B
0.003	0.0	0.0	0.0	0.003	0.004	0.003	0.005	0.005	0.004
INT	CZZ	MMP	CVI	BP	CVX	CEO	XOM	OKE	PTR
0.003	0.004	0.002	0.003	0.003	0.004	0.004	0.003	0.002	0.004
RNR	WTM	TPVG	JPM	RFI	SC	BBN	HIG	NVG	WFC
0.003	0.003	0.004	0.004	0.004	0.004	0.003	0.003	0.005	0.004
SAR	ALL	BTT	PMT	DFP	MET	PNC	RGA		
0.004	0.002	0.004	0.003	0.003	0	0	0.003		

Table 6.17: Genetic algorithm portfolio for NYSE stock exchange

6.4 Out-of-sample Validation

Generally, the comparison of the output of the proposed methodological framework to the market returns is a necessary procedure in order to evaluate the accuracy of the methodology. This procedure is vital for the confirmation of the produced output. It is obvious that the comparison must take place after the moment of the investment. Therefore, the goal of the validation is to prove that the produced portfolios perform equally good or better compared to the market indexes, for a period after the analysis time horizon.

Therefore, the validation process is based on out-of-sample data, which do not belong to the initial set of input data which were used during the analysis. Thus, the portfolio optimisation procedure was based on daily data for the time period from 01/01/2016 until 31/12/2018. The validation process takes place in the following time period, from 01/01/2019 until 30/06/2019. The comparison was conducted for three different time periods: (a) short-term (1 month), (b) mid-term (3 months), (c) long-term (6 months).

NYSE Stock Exchange The results of the validation process for NYSE stock exchange are presented in table 6.18. Additionally, figure 6.12 includes a comparative graphical representation of the expected daily capital return (%) for the selected optimal portfolio with every optimisation model compared to the market index. The empirical testing procedure for NYSE stock exchange resulted in the following findings:

- During the period of January 2019 (1 month time horizon), all four models perform better than NYSE market index. The PROMETHEE flow model seems to offer the best expected return compared to the other models, while the MIQP mean - variance model offers the lowest expected return.

- During the period of January-March 2019 (3 month time horizon), the genetic algorithm model seems to perform better than the other models. However, it is important to note that all the models offer a higher expected return compared to the market index.

- Finally, during the period of January-June 2019 (6 month time horizon), all four models offer similar expected return, which is significantly higher than the market index return. Among the four models, the MIQP mean - variance model seems to offer slightly better results for a long term.

Time Horizon	Market Index	MIQP Mean – Variance	Goal Programming	MOIP PROMETHEE flow	Genetic Algorithm
1 month	0.4048	0.441	0.4874	0.6948	0.468
3 months	0.2049	0.2648	0.2836	0.2932	0.3135
6 months	0.1182	0.2565	0.2336	0.2353	0.2119

Table 6.18: Expected daily capital return (%) for selected optimal portfolios and market index (NYSE stock exchange)

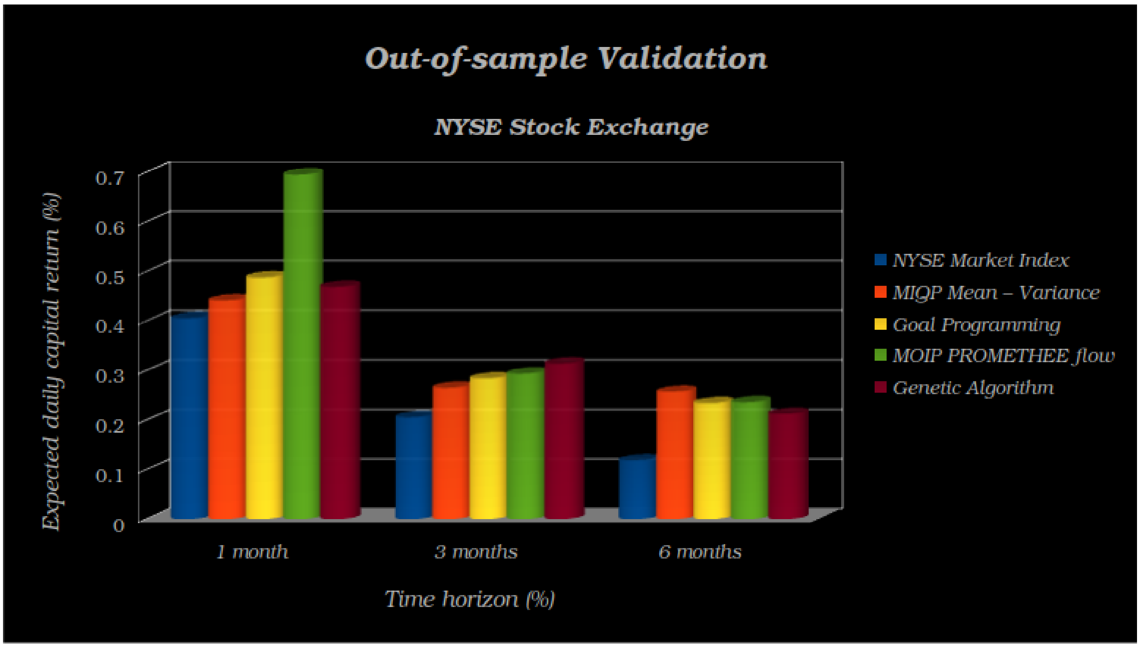


Figure 6.12: Comparative graphical representation of expected daily capital return (%) for selected optimal portfolio and market index (NYSE stock exchange)

NASDAQ Stock Exchange The results of the validation process for NASDAQ stock exchange are presented in table 6.19. Additionally, figure 6.13 includes a comparative graphical representation of the expected daily capital return (%) for the selected optimal portfolio with every optimisation model compared to the market index. The empirical testing procedure for NASDAQ stock exchange resulted in the following findings:

- During the period of January 2019 (1 month time horizon), we observe that all four models perform better than NASDAQ market index. The goal programming model seems to offer the best expected return compared to the other models, while the MIQP mean - variance model offers the lowest expected return.
- During the period of January-March 2019 (3 month time horizon), we note that all four models offer significantly better return than NASDAQ market index. Among the four models, the genetic algorithm model seems to perform worse

than the other models, while the other three models offer similar expected return.

- Finally, during the period of January-June 2019 (6 month time horizon), it is obvious that the goal programming model offers the best result, while the PROMETHEE flow model performs equally well. As in the previous cases, all four models are more profitable than the market index.

Time Horizon	Market Index	MIQP Mean – Variance	Goal Programming	MOIP PROMETHEE flow	Genetic Algorithm
1 month	0.4612	0.3318	0.2924	0.3719	0.3456
3 months	0.2914	0.1996	0.2938	0.2861	0.2807
6 months	0.1712	0.1904	0.2081	0.2766	0.1847

Table 6.19: Expected daily capital return (%) for selected optimal portfolios and market index (NASDAQ stock exchange)

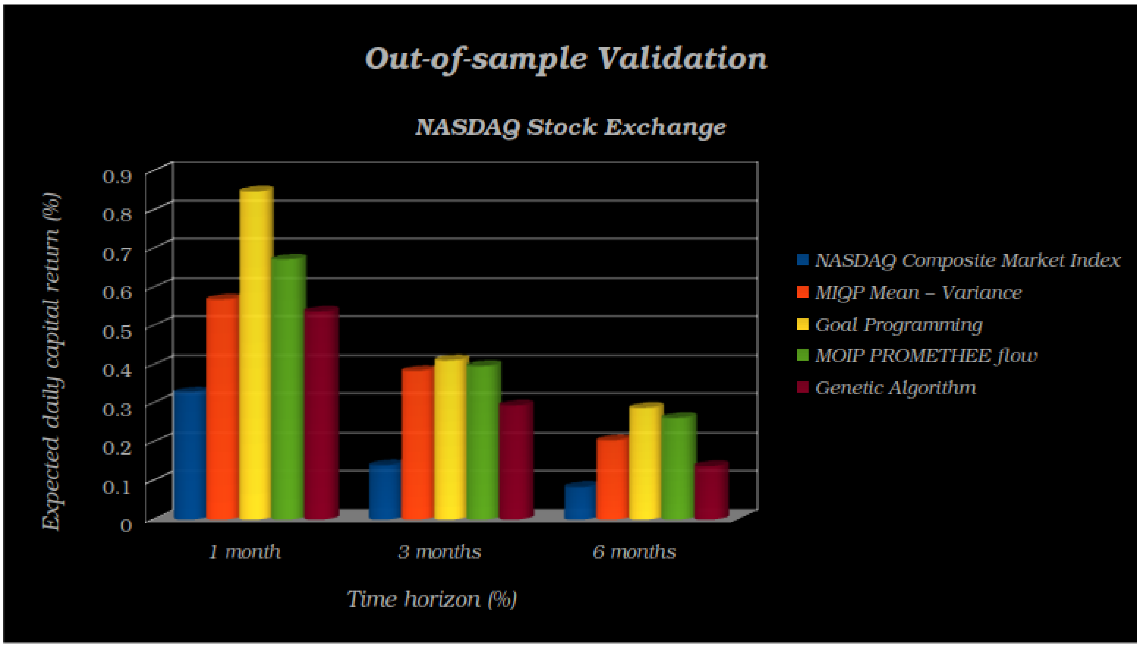


Figure 6.13: Comparative graphical representation of expected daily capital return (%) for selected optimal portfolio and market index (NASDAQ stock exchange)

PARIS Stock Exchange The results of the validation process for Paris stock exchange are presented in table 6.20. Additionally, figure 6.14 includes a comparative graphical representation of the expected daily capital return (%) for the selected optimal portfolio with every optimisation model compared to the market index. The empirical testing procedure for Paris stock exchange resulted in the following findings:

- During the period of January 2019 (1 month time horizon), only the goal programming model performs better than CAC40 market index, while the genetic programming model performs equally well. On the other hand the PROMETHEE flow and the mean - variance models seem to offer lower expected return compared to the market.

- During the period of January-March 2019 (3 month time horizon), the goal programming model seems to perform better than all the other models, while the genetic algorithm model offers similar expected return compared to the market index.

- Finally, during the period of January-June 2019 (6 month time horizon), the genetic algorithm model offers the greatest expected return. The only model that performs slightly worse than the market index is the MIQP mean - variance model.

Time Horizon	**Market Index**	**MIQP Mean – Variance**	**Goal Programming**	**MOIP PROMETHEE flow**	**Genetic Algorithm**
1 month	0.3517	0.2327	0.3864	0.2357	0.3384
3 months	0.2285	0.1852	0.3118	0.2355	0.1721
6 months	0.1513	0.0938	0.2391	0.1972	0.2692

Table 6.20: Expected daily capital return (%) for selected optimal portfolios and market index (PARIS stock exchange)

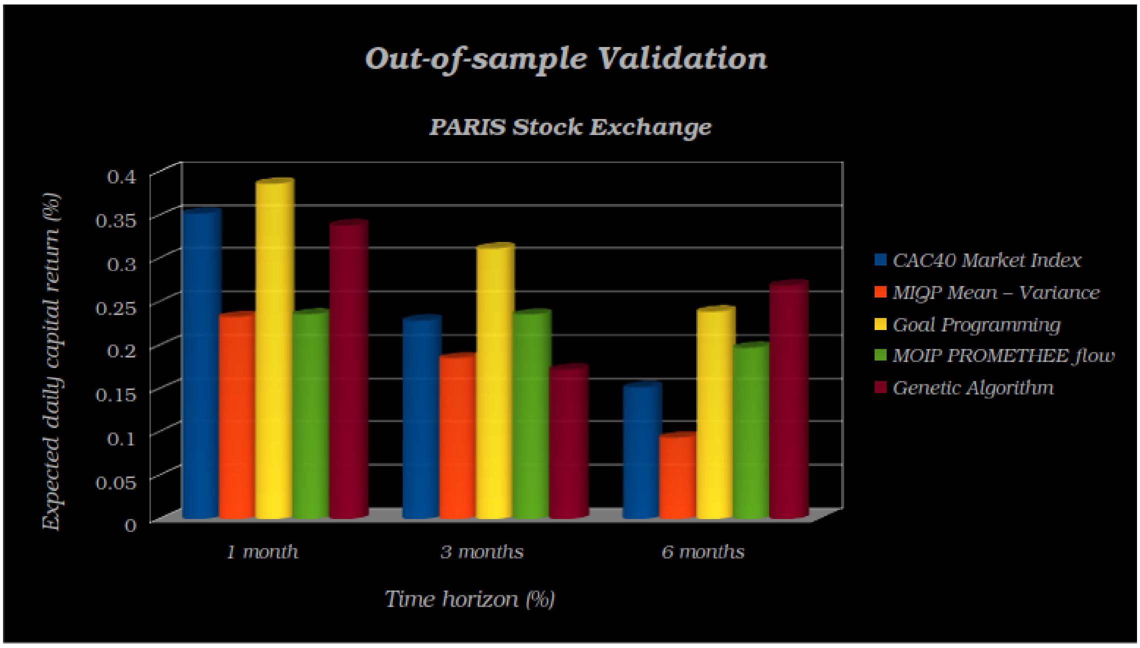


Figure 6.14: Comparative graphical representation of expected daily capital return (%) for selected optimal portfolio and market index (PARIS stock exchange)

TOKYO Stock Exchange The results of the validation process for Tokyo stock exchange are presented in table 6.21. Additionally, figure 6.15 includes a comparative graphical representation of the expected daily capital return (%) for the selected optimal portfolio with every optimisation model compared to the market index. The empirical testing procedure for Tokyo stock exchange resulted in the following findings:

- During the period of January 2019 (1 month time horizon), all four models perform better than Nikkei225 market index. The goal programming model seems to offer the best expected return compared to the other models, while the MIQP mean - variance and the genetic algorithm model offer the lowest expected return.

- During the period of January-March 2019 (3 month time horizon), all four models continue to perform better than Nikkei225 market index. More specifically, the goal programming model seems to perform slightly better than the other models, while the genetic algorithm model offers the lowest expected return.

- Finally, during the period of January-June 2019 (6 month time horizon), all four models offer higher expected return compared to the market index. Among the four models, the goal programming model seems to offer slightly better results for a long term.

Time Horizon	**Market Index**	**MIQP Mean − Variance**	**Goal Programming**	**MOIP PROMETHEE flow**	**Genetic Algorithm**
1 month	0.3277	0.5681	0.8468	0.6715	0.5368
3 months	0.139	0.3829	0.4103	0.3963	0.2942
6 months	0.0829	0.2032	0.288	0.2616	0.1368

Table 6.21: Expected daily capital return (%) for selected optimal portfolios and market index (TOKYO stock exchange)

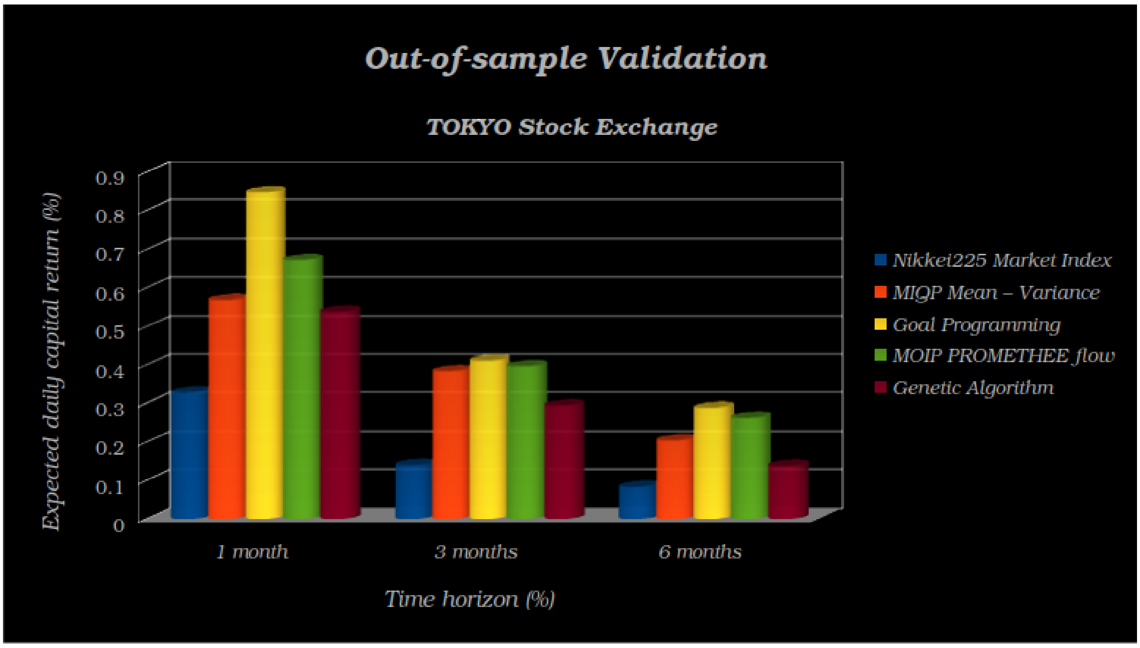


Figure 6.15: Comparative graphical representation of expected daily capital return (%) for selected optimal portfolio and market index (TOKYO stock exchange)

Conclusively, the out-of-sample validation certifies the validity of the methodological framework. As stated in the previous paragraph the constructed portfolios perform equally good or even better than the market index, thus rendering the proposed methodology a reliable tool in the hands of the decision-maker.

Chapter 7

Conclusion - Future Prospects

7.1 Conclusion

Nowadays, the necessity for integrated methodological frameworks and decision support systems is stronger than ever. These methodologies should encapsulate all the conflicting criteria and the interactions among them, as well as the uncertainty of the financial market and the different profiles of the decision-makers. Consequently, this thesis has contributed to the recognition of all the parameters of the portfolio management problem and the interactions among all these factors. In this section, we present the main conclusions of the previous chapters.

The fundamental mean-variance model is based on two criteria (return and risk), which renders it insufficient to deal with a realistic problem. On the contrary, an integrated approach demands the incorporation of all the conflicting criteria that affect the security market. Consequently, the portfolio management problem is a multiple-criteria problem which demands a multicriteria integrated methodological framework.

The related methodologies overview leads to the conclusion that the multicriteria decision analysis is the most suitable tool to use against the problem of portfolio management. However, the existing methodologies have some major disadvantages: (i) There are only few integrated methodologies that deal with problem of portfolio management in a systemic and complete way and (ii) the proposed methodologies involve large computational effort (even larger than the mean-variance model) thus rendering them useless in the case that there is a large number of securities.

Having recognised all the difficulties of the problem and the defficiencies in the existing methodologies, the current thesis involves with the development of a complete methodological framework for portfolio management. This framework consists of two major phases: (i) The multicriteria portfolio selection phase and (ii) the multiobjective optimisation phase. These two phases can be applied separately, but they can also be used sequentially constituting an integrated methodological framework.

The process of portfolio management emphasises the necessity for modern information system which implement the theoritical models efficiently. Therefore, as part of this project an information system was designed which implements the various methods. The programming language which was used for the whole implementation is Python 3, accompanied with some scientific libraries such as matplotlib, numpy and pandas. Additionally, the MCDA portfolio selection subsystem was developped as a web application with a user-friendly user interface.

The proposed methodology was succesfully tested in four stock exchanges and in three industrial sectors. The computational effort was significantly lower compared to the conventional methods and the results of the proposed methodology in the out-of-sample validation were very encouraging.

7.2 Future Prospects

The current thesis creates some future prospects of research on the portfolio management problem. The major future prospects are the following:

Deployment of the whole information system as a web application: As part of the project, an integrated decision support information system is implemented. Additionally, the MCDA portfolio selection subsystem is deployed as web application. An important future prospect in this field is the deployment of the whole system as a web application. The implementation of the whole system deployment is not of great difficulty, as the development of the other subsystems is based on a generalisation of the existing application.

Connection of the information system to commercial applications: The developed information system implements the proposed portfolio management methodology, constituting an integrated decision support tool. The future prospect in this scientific field includes the connection of the information system to existing commercial applications. This connection would facilitate the decision support procedure for each customer. However, this innovative prospect would lead to a series of application security issues.

Appendix A

Source Code Presentation

In this appendix we present the source code of the information system. The appendix is organised in two paragraphs. In the first paragraph, there is a detailed explanation of the code that implements the MCDA methods for portfolio selection, whereas in the second paragraph there is a presentation of the multiobjective portfolio optimisation techniques. The explanation of the source code is made with Jupiter Notebook according to the presentation that was made in the main body of the thesis.

Phase I: Multicriteria Portfolio Selection

The input data that is used in order to explain the source code is based an original dataset of the experimental application. More specifically, it includes financial indexes for the available securities of NYSE stock exchange that belong to the technology industrial sector.

[1]:
```
import csv
import numpy as np
import matplotlib.pyplot as plt
import fpdf
import math
import pandas as pd
```

After importing all necessary libraries we should define the input .csv file that includes the financial indexes. This file is formulated from data fetched by the investing.com database and it contains the evaluation table for all criteria for each alternative.

[2]:
```
inputname = "/home/elissaios/Documents/Thesis/1) MCDA/1.NYSE/
↪NYSE_TECHNOLOGY.csv"
```

```
file =  open(inputname, "rt")
list1 = list(csv.reader(file))
```

Now, list1 contains all the information of the experiment of MCDA portfolio selection. Therefore, the next step involves taking all the useful data from this list.

[3]:
```
criteria = int(list1[0][1])
alternatives = int(list1[1][1])
weights = [0 for y in range(criteria)]
optimizationType = [0 for y in range(criteria)]
preferenceThreshold = [0 for y in range(criteria)]
criterion = [0 for y in range(criteria)]
indifferenceThreshold = [0 for y in range(criteria)]
criterionName = [0 for y in range(criteria)]
vetoThreshold = [0 for y in range(criteria)]
for i in range(criteria):
    optimizationType[i] = int(list1[2][i+1])
    weights[i] = float(list1[3][i+1])
    vetoThreshold[i] = float(list1[4][i+1])
    criterion[i] = int(list1[7][i+1])
    criterionName[i] = list1[9][i+1]
    preferenceThreshold[i] = float(list1[5][i+1])
    indifferenceThreshold[i] = float(list1[6][i+1])
decisionMatrix = [[0 for y in range(criteria)] for x in␣
 ↪range(alternatives)]
companyName = ["" for i in range(alternatives)]
for i in range(alternatives):
    companyName[i] = list1[i+10][0]
    for j in range(criteria):
        decisionMatrix[i][j] = float(list1[i+10][j+1])
```

In the following cell we print the decision matrix that we have taken from the input list, in a dataframe form using pandas function dataframe as follows:

[4]:
```
print("=============== Decision Matrix ===============\n")
df = pd.DataFrame.from_records(decisionMatrix, index=companyName␣
 ↪,columns=criterionName)
display(df)
```

```
======================= Decision Matrix =======================
```

```
                      P/E Ratio   EPS  Revenue (B)   Beta  Dividend Yield (%)  \
ABB ADR                   43.75  0.41       41.680   1.19                4.33
Accenture                 25.05  7.36       43.220   1.05                1.74
SAP ADR                   40.20  2.87       29.300   1.11                1.45
Infosys ADR               22.10  0.50       12.300   0.48                2.20
Wipro ADR                 16.74  0.22        8.550   0.50                0.29
...                         ...   ...          ...    ...                 ...
Taiwan Semiconductor      24.51  2.00       32.670   0.97                3.26
Servotronics               7.20  1.37        0.051   0.53                1.62
Synnex                    12.63  8.91       22.800   0.99                1.33
TransUnion                49.59  1.61        2.500   0.99                0.38
Xerox                     12.01  2.40        9.380   1.80                3.48

                      Monthly  YTD  (%)  1 Year
ABB ADR                   1.0     -4.52  -14.63
Accenture                 5.0     30.37   15.30
SAP ADR                   3.0     15.77    0.63
Infosys ADR               5.0     17.44   14.78
Wipro ADR                 1.0     -5.71   -4.22
...                       ...       ...     ...
Taiwan Semiconductor      5.0     32.57   23.37
Servotronics              2.0     -0.60   -4.90
Synnex                    5.0     39.26   51.26
TransUnion                5.0     40.51   15.77
Xerox                     4.0     45.85   14.27

[69 rows x 8 columns]
```

After we have translated all the data from the .csv file that was imported, we present the source code of the implementation of the 4 MCDA ranking methods for portfolio selection.

MAUT

Initially, we should calculate the normalised decision matrix:

[5]:
```
maxValue = np.max(decisionMatrix, axis = 0)
minValue = np.min(decisionMatrix, axis = 0)

normalisedMatrix = [[0 for y in range(criteria)] for x in
 ↪range(alternatives)]
for i in range(alternatives):
        for j in range(criteria):
                if optimizationType[j] == 0:
```

```
                        normalisedMatrix[i][j] =␣
↪(decisionMatrix[i][j] - minValue[j])*1.0 / (maxValue[j] -␣
↪minValue[j])
                elif optimizationType[j] == 1:
                        normalisedMatrix[i][j] = (maxValue[j] -␣
↪decisionMatrix[i][j])*1.0 / (maxValue[j] - minValue[j])

print("=============== Normalised decision Matrix␣
↪===============\n")
df = pd.DataFrame.from_records(normalisedMatrix, index=companyName␣
↪,columns=criterionName)
display(df)
```

```
==================== Normalised decision Matrix ====================


                      P/E Ratio       EPS  Revenue (B)      Beta  \
ABB ADR                0.680573  0.018299     0.535230  0.611336
Accenture              0.827771  0.362018     0.555013  0.668016
SAP ADR                0.708517  0.139960     0.376196  0.643725
Infosys ADR            0.850992  0.022750     0.157814  0.898785
Wipro ADR              0.893183  0.008902     0.109641  0.890688
...                         ...       ...          ...       ...
Taiwan Semiconductor   0.832021  0.096934     0.419487  0.700405
Servotronics           0.968278  0.065776     0.000462  0.878543
Synnex                 0.925535  0.438675     0.292697  0.692308
TransUnion             0.634603  0.077646     0.031922  0.692308
Xerox                  0.930416  0.116716     0.120303  0.364372

                      Dividend Yield (%)  Monthly  YTD  (%)    1 Year
ABB ADR                         0.450390     0.00  0.202188  0.201037
Accenture                       0.161650     1.00  0.357330  0.310359
SAP ADR                         0.129320     0.50  0.292410  0.256776
Infosys ADR                     0.212932     1.00  0.299835  0.308459
Wipro ADR                       0.000000     0.00  0.196896  0.239061
...                                  ...      ...       ...       ...
Taiwan Semiconductor            0.331104     1.00  0.367113  0.339835
Servotronics                    0.148272     0.25  0.219618  0.236577
Synnex                          0.115942     1.00  0.396861  0.441705
TransUnion                      0.010033     1.00  0.402419  0.312075
Xerox                           0.355630     0.75  0.426164  0.306597

[69 rows x 8 columns]
```

It is obvious that the values of the initial decision matrix have been normalised to 0-1. After the normalisation process the utility score is calculated, as follows:

[6]:
```
utilityScore = [0 for x in range(alternatives)]
utilityScorePer = [0 for x in range(alternatives)]

for i in range(alternatives):
        tempSum = 0
        for j in range(criteria):
                tempSum += normalisedMatrix[i][j] * weights[j]
        utilityScore[i] = round(tempSum,4)
        utilityScorePer[i] = round(round(tempSum,4) * 100,2)

print("=============== Utility Score ===============\n")
df = pd.DataFrame(utilityScore, index=companyName,␣
 ↪columns=["Score"])
display(df)
```

```
======================= Utility Score =======================



                      Score
ABB ADR              0.2699
Accenture            0.6242
SAP ADR              0.4047
Infosys ADR          0.5752
Wipro ADR            0.2338
...                     ...
Taiwan Semiconductor  0.6087
Servotronics         0.3268
Synnex               0.6304
TransUnion           0.5161
Xerox                0.4870

[69 rows x 1 columns]
```

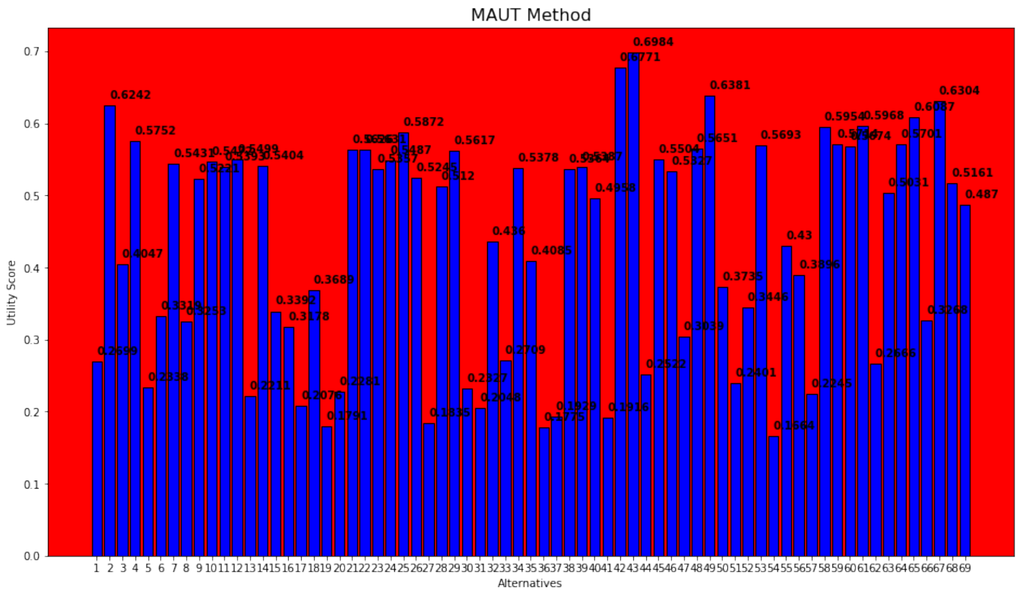
Finally, the results should be sorted in order to find the ranking of the method. The following shell describes this process and offers a visualisation of the results.

[7]:
```
tupledList = list(zip(companyName,utilityScorePer))
tupledListSorted = sorted(tupledList, key=lambda tup: tup[1],␣
 ↪reverse=True)
```

```
print("=============== MAUT Ranking ===============\n")
df = pd.DataFrame(tupledListSorted, columns=["Company", "Score"])
df.index = df.index + 1
display(df)

i = np.arange(alternatives)
plt.figure(figsize=(16,9))
plt.bar(i+1, utilityScore, color = 'b', edgecolor = 'black')
plt.xlabel('Alternatives')
plt.ylabel('Utility Score')
plt.title('MAUT Method', fontsize=16)
ax = plt.gca()
ax.set_facecolor('red')
xlocs, xlabs = plt.xticks(np.arange(1, alternatives+1, step=1))
xlocs=[i for i in range(1,alternatives+1)]
xlabs=[i/2 for i in range(1,alternatives+1)]
# plt.xticks(xlocs, xlabs)
for i, v in enumerate(utilityScore):
    plt.text(xlocs[i] -0.5/alternatives, v + 0.01, str(v),␣
 ↪fontweight='bold')

plt.show()
```

```
==================== MAUT Ranking ====================
```

```
              Company  Score
1     Northrop Grumman  69.84
2          GlobalSCAPE  67.71
3                  IBM  63.81
4              Synnex  63.04
5           Accenture  62.42
..                ...    ...
65             Fortive  19.16
66  Issuer Direct Corp  18.35
67              Belden  17.91
68           Energizer  17.75
69        Yirendai Ltd  16.64

[69 rows x 2 columns]
```



TOPSIS

In TOPSIS method we initially calculate the normalised decision matrix as you can see in the following cell:

[8]:
```
normalisedDecisionMatrix = [[0 for i in range(criteria)] for y in␣
↪range(alternatives)]
for j in range(criteria):
    sumOfPows = 0
    for i in range(alternatives):
        sumOfPows = sumOfPows + math.pow(decisionMatrix[i][j],2)
        sqSumOfPows =  math.sqrt(sumOfPows)
        for i in range(alternatives):
            normalisedDecisionMatrix[i][j] =␣
↪round(decisionMatrix[i][j]*1.0 / sqSumOfPows,3)

print("=============== Normalised decision Matrix␣
↪===============\n")
df = pd.DataFrame.from_records(normalisedDecisionMatrix,␣
↪index=companyName ,columns=criterionName)
display(df)
```

```
==================== Normalised decision Matrix ====================

                      P/E Ratio    EPS  Revenue (B)    Beta  \
```

```
ABB ADR                  0.164  0.010        0.267  0.116
Accenture                0.094  0.187        0.277  0.103
SAP ADR                  0.150  0.073        0.188  0.109
Infosys ADR              0.083  0.013        0.079  0.047
Wipro ADR                0.063  0.006        0.055  0.049
...                        ...    ...          ...    ...
Taiwan Semiconductor     0.092  0.051        0.210  0.095
Servotronics             0.027  0.035        0.000  0.052
Synnex                   0.047  0.226        0.146  0.097
TransUnion               0.185  0.041        0.016  0.097
Xerox                    0.045  0.061        0.060  0.176

                      Dividend Yield (%)  Monthly   YTD  (%)  1 Year
ABB ADR                            0.208     0.032     -0.016  -0.052
Accenture                          0.083     0.158      0.107   0.054
SAP ADR                            0.070     0.095      0.056   0.002
Infosys ADR                        0.106     0.158      0.061   0.052
Wipro ADR                          0.014     0.032     -0.020  -0.015
...                                  ...       ...        ...     ...
Taiwan Semiconductor               0.156     0.158      0.115   0.082
Servotronics                       0.078     0.063     -0.002  -0.017
Synnex                             0.064     0.158      0.138   0.181
TransUnion                         0.018     0.158      0.143   0.056
Xerox                              0.167     0.126      0.161   0.050

[69 rows x 8 columns]
```

In the following step we incorporate the offset, computing the weight decision matrix:

[9]:
```
weightedDecisionMatrix = [[0 for i in range(criteria)] for y in␣
 ↪range(alternatives)]
for j in range(criteria):
    for i in range(alternatives):
        weightedDecisionMatrix[i][j] =␣
 ↪round(normalisedDecisionMatrix[i][j] * weights[j],3)

print("=============== Weighted Decision Matrix ===============\n")
df = pd.DataFrame.from_records(weightedDecisionMatrix,␣
 ↪index=companyName ,columns=criterionName)
display(df)
```

```
================= Weighted Decision Matrix =================


                      P/E Ratio    EPS  Revenue (B)    Beta  \
```

```
ABB ADR                   0.016  0.001       0.027  0.012
Accenture                 0.009  0.019       0.028  0.010
SAP ADR                   0.015  0.007       0.019  0.011
Infosys ADR               0.008  0.001       0.008  0.005
Wipro ADR                 0.006  0.001       0.006  0.005
...                         ...    ...         ...    ...
Taiwan Semiconductor      0.009  0.005       0.021  0.010
Servotronics              0.003  0.004       0.000  0.005
Synnex                    0.005  0.023       0.015  0.010
TransUnion                0.018  0.004       0.002  0.010
Xerox                     0.004  0.006       0.006  0.018

                      Dividend Yield (%)  Monthly  YTD  (%)  1 Year
ABB ADR                            0.021    0.010    -0.002  -0.005
Accenture                          0.008    0.047     0.011   0.005
SAP ADR                            0.007    0.028     0.006   0.000
Infosys ADR                        0.011    0.047     0.006   0.005
Wipro ADR                          0.001    0.010    -0.002  -0.002
...                                  ...      ...       ...     ...
Taiwan Semiconductor               0.016    0.047     0.012   0.008
Servotronics                       0.008    0.019    -0.000  -0.002
Synnex                             0.006    0.047     0.014   0.018
TransUnion                         0.002    0.047     0.014   0.006
Xerox                              0.017    0.038     0.016   0.005

[69 rows x 8 columns]
```

Subsequently, we calculate the positive and negative ideal solutions as well as the distance of each alternative from them:

[10]:
```
idealSolution = [0 for i in range(criteria)]
nonIdealSolution = [0 for i in range(criteria)]
for j in range(criteria):
    maxValue = float('-inf')
    minValue = float('inf')
    for i in range(alternatives):
        if weightedDecisionMatrix[i][j] < minValue:
            minValue = weightedDecisionMatrix[i][j]
        if weightedDecisionMatrix[i][j] > maxValue:
            maxValue = weightedDecisionMatrix[i][j]
    if optimizationType[j] == 0:
        idealSolution[j] = maxValue
        nonIdealSolution[j] = minValue
    elif optimizationType[j] == 1:
        idealSolution[j] = minValue
        nonIdealSolution[j] = maxValue
```

```
sPlus = [0 for i in range(alternatives)]
sMinus = [0 for i in range(alternatives)]
for i in range(alternatives):
    sumPlusTemp = 0
    sumMinusTemp = 0
    for j in range(criteria):
        sumPlusTemp = sumPlusTemp + math.
 ↪pow(idealSolution[j]-weightedDecisionMatrix[i][j],2)
        sumMinusTemp = sumMinusTemp + math.
 ↪pow(nonIdealSolution[j]-weightedDecisionMatrix[i][j],2)
    sPlus[i] = math.sqrt(sumPlusTemp)
    sMinus[i] = math.sqrt(sumMinusTemp)

print("=============== Positive Ideal Solution ===============\n")
df = pd.DataFrame(idealSolution, index=criterionName,␣
 ↪columns=["Score"])
display(df)
print("\n")
print("=============== Negative Ideal Solution ===============\n")
df = pd.DataFrame(nonIdealSolution, index=criterionName,␣
 ↪columns=["Score"])
display(df)
print("\n")
print("=============== Distance from Positive Ideal Solution␣
 ↪===============\n")
df = pd.DataFrame(sPlus, index=companyName, columns=["Distance"])
display(df)
print("\n")
print("=============== Distance from Negative Ideal Solution␣
 ↪===============\n")
df = pd.DataFrame(sMinus, index=companyName, columns=["Distance"])
display(df)
```

```
================== Positive Ideal Solution ==================


                    Score
P/E Ratio           0.001
EPS                 0.051
Revenue (B)         0.050
Beta                0.002
Dividend Yield (%)  0.044
Monthly             0.047
YTD  (%)            0.062
1 Year              0.072

================== Negative Ideal Solution ==================
```

```
                     Score
P/E Ratio            0.049
EPS                  0.000
Revenue (B)          0.000
Beta                 0.026
Dividend Yield (%)   0.001
Monthly              0.010
YTD  (%)            -0.018
1 Year              -0.025

=============== Distance from Positive Ideal Solution ===============

                       Distance
ABB ADR                0.123600
Accenture              0.100110
SAP ADR                0.114996
Infosys ADR            0.114175
Wipro ADR              0.131377
...                         ...
Taiwan Semiconductor   0.102299
Servotronics           0.126972
Synnex                 0.093557
TransUnion             0.115282
Xerox                  0.107893

[69 rows x 1 columns]

=============== Distance from Negative Ideal Solution ===============

                       Distance
ABB ADR                0.055417
Accenture              0.078486
SAP ADR                0.057896
Infosys ADR            0.071638
Wipro ADR              0.055785
...                         ...
Taiwan Semiconductor   0.076844
Servotronics           0.059632
Synnex                 0.084929
TransUnion             0.067764
Xerox                  0.072505

[69 rows x 1 columns]
```

Finally, we calculate the relative closeness of every alternative to the ideal solution:

[11]:
```
C = [0 for i in range(alternatives)]
C2 = [0 for i in range(alternatives)]
for i in range(alternatives):
    C2[i] = round(round(sMinus[i]*1.0 / (sMinus[i] + sPlus[i]),4)␣
 ↪* 100,2) #percentage
    C[i] = sMinus[i]*1.0 / (sMinus[i] + sPlus[i])

print("=============== Relative Closeness ===============\n")
df = pd.DataFrame(C, index=companyName, columns=["Distance"])
display(df)
```

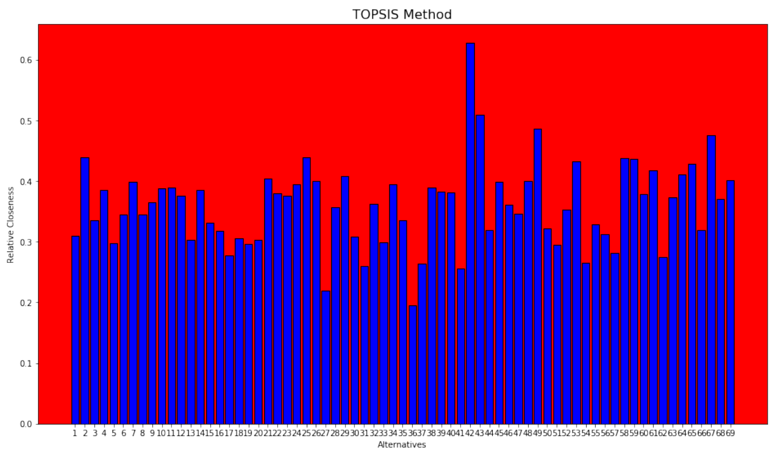
```
======================= Relative Closeness =======================


                      Distance
ABB ADR               0.309561
Accenture             0.439460
SAP ADR               0.334870
Infosys ADR           0.385537
Wipro ADR             0.298058
...                        ...
Taiwan Semiconductor  0.428954
Servotronics          0.319564
Synnex                0.475830
TransUnion            0.370203
Xerox                 0.401916

[69 rows x 1 columns]
```

The final ranking of TOPSIS method is provided if we sort the relative closeness. In the following cell we describe this procedure as well the visualisation of the results.

[12]:
```
tupledList = list(zip(companyName,C))
tupledListSorted = sorted(tupledList, key=lambda tup: tup[1],␣
 ↪reverse=True)

print("=============== TOPSIS Ranking ===============\n")
df = pd.DataFrame(tupledListSorted, columns=["Company", "Score"])
df.index = df.index + 1
display(df)

i = np.arange(alternatives)
plt.figure(figsize=(16,9))
```

```
plt.bar(i+1, C, color = 'b', edgecolor = 'black')
plt.xlabel('Alternatives')
plt.ylabel('Relative Closeness')
plt.title('TOPSIS Method', fontsize=16)
ax = plt.gca()
ax.set_facecolor('red')
xlocs, xlabs = plt.xticks(np.arange(1, alternatives+1, step=1))
xlocs=[i for i in range(1,alternatives+1)]
xlabs=[i/2 for i in range(1,alternatives+1)]
# plt.xticks(xlocs, xlabs)


plt.show()
```

```
======================= TOPSIS Ranking =======================



                 Company     Score
1             GlobalSCAPE  0.628233
2       Northrop Grumman  0.508793
3                    IBM  0.486176
4                 Synnex  0.475830
5              Accenture  0.439460
..                   ...       ...
65              Enersys  0.263751
66        DXC Technology  0.259741
67               Fortive  0.255952
68    Issuer Direct Corp  0.219171
69             Energizer  0.194777

[69 rows x 2 columns]
```



ELECTRE III

ELECTRE III belongs to the ELECTRE family methods which are based on a different concept than the previous ones. This method is based on pairwise comparisons between the alternatives. Firstly, we calculate the agreement and the disagreement tables:

[13]:
```
sumOfWeights = sum(weights)
agreementTable = [[0 for i in range(alternatives)] for y in␣
 ↪range(alternatives)]
for k in range(criteria):
    if optimizationType[k] == 0:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j:
                    if decisionMatrix[j][k] - decisionMatrix[i][k]␣
 ↪<= indifferenceThreshold[k]:
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + 1.0 * weights[k],2)
                    elif decisionMatrix[j][k] -␣
 ↪decisionMatrix[i][k] <= preferenceThreshold[k]:
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + ((decisionMatrix[i][k] -␣
 ↪decisionMatrix [j][k] + preferenceThreshold[k])*1.0 /␣
 ↪(preferenceThreshold[k]-indifferenceThreshold[k])) *␣
 ↪weights[k],2)
                    else:
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + 0.0 * weights[k],2)
    elif optimizationType[k] == 1:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j:
                    if decisionMatrix[i][k] - decisionMatrix[j][k]␣
 ↪<= indifferenceThreshold[k]:
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + 1.0 * weights[k],2)
                    elif decisionMatrix[i][k] -␣
 ↪decisionMatrix[j][k] <= preferenceThreshold[k]:
```

```
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + ((decisionMatrix[j][k] -␣
 ↪decisionMatrix [i][k] + preferenceThreshold[k])*1.0 /␣
 ↪(preferenceThreshold[k]-indifferenceThreshold[k])) *␣
 ↪weights[k],2)
                    else:
                        agreementTable[i][j] =␣
 ↪round(agreementTable[i][j] + 0.0 * weights[k],2)

disagreementTable = [[[0 for k in range(criteria)] for i in␣
 ↪range(alternatives)] for j in range(alternatives)]
for k in range(criteria):
    if optimizationType[k] == 0:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j:
                    if decisionMatrix[j][k] - decisionMatrix[i][k]␣
 ↪<= preferenceThreshold[k]:
                        disagreementTable[i][j][k] = 0
                    elif decisionMatrix[j][k] -␣
 ↪decisionMatrix[i][k] <= vetoThreshold[k]:
                        disagreementTable[i][j][k] =␣
 ↪round(((decisionMatrix[j][k] - decisionMatrix [i][k] -␣
 ↪preferenceThreshold[k])*1.0 /␣
 ↪(vetoThreshold[k]-preferenceThreshold[k])),2)
                    else:
                        disagreementTable[i][j][k] = 1
    elif optimizationType[k] == 1:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j:
                    if decisionMatrix[i][k] - decisionMatrix[j][k]␣
 ↪<= indifferenceThreshold[k]:
                        disagreementTable[i][j][k] = 0
                    elif decisionMatrix[i][k] -␣
 ↪decisionMatrix[j][k] <= vetoThreshold[k]:
                        disagreementTable[i][j][k] =␣
 ↪round(((decisionMatrix[j][k] - decisionMatrix [i][k] +␣
 ↪preferenceThreshold[k])*1.0 /␣
 ↪(vetoThreshold[k]-preferenceThreshold[k])),2)
                    else:
                        disagreementTable[i][j][k] = 1
```

In the following step we are able to calculate the reliability and dominance tables based on the above tables:

[14]:
```
reliabilityTable = [[0 for i in range(alternatives)] for y in␣
↪range(alternatives)]
for i in range(alternatives):
    for j in range(alternatives):
        if i!=j:
            reliabilityTable[i][j] = agreementTable[i][j]
            for k in range(criteria):
                if agreementTable[i][j] <␣
↪disagreementTable[i][j][k]:
                    reliabilityTable[i][j] =␣
↪round(reliabilityTable[i][j] * ((1 - disagreementTable[i][j][k])␣
↪/ (1 - agreementTable[i][j])), 2)

d = 0.8
dominanceTable = [[0 for i in range(alternatives)] for y in␣
↪range(alternatives)]
for i in range(alternatives):
    for j in range(alternatives):
        if i!=j and reliabilityTable[i][j] >= d:
            dominanceTable[i][j] = 1
```

Finally, the proposed version of ELECTRE III suggests the calculation of the positive and negative flow for each alternative. Based on these two, we can compute the final ELECTRE III flow, as described in the following cell:

[15]:
```
phiPlus = [round(sum(x),6) for x in reliabilityTable ]
phiMinus = [round(sum(x),6) for x in zip(*reliabilityTable)]
phiEl = [round(x1 - x2,6) for (x1, x2) in zip(phiPlus, phiMinus)]

print("=============== Positive Flow ===============\n")
df = pd.DataFrame(phiPlus, index=companyName, columns=["Flow"])
display(df)
print("\n")
print("=============== Negative Flow ===============\n")
df = pd.DataFrame(phiMinus, index=companyName, columns=["Flow"])
display(df)
print("\n")
print("=============== ELECTRE III Flow ===============\n")
df = pd.DataFrame(phiEl, index=companyName, columns=["Flow"])
display(df)
```

```
=============== Positive Flow ===============
```

```
                      Flow
ABB ADR              26.06
Accenture            59.32
SAP ADR              50.52
Infosys ADR          53.95
Wipro ADR            22.68
...                    ...
Taiwan Semiconductor  61.14
Servotronics         38.14
Synnex               60.69
TransUnion           49.79
Xerox                41.90

[69 rows x 1 columns]




=============== Negative Flow ===============



                      Flow
ABB ADR              14.49
Accenture             7.66
SAP ADR              48.78
Infosys ADR          35.20
Wipro ADR            51.57
...                    ...
Taiwan Semiconductor  18.25
Servotronics         52.72
Synnex               26.64
TransUnion           41.05
Xerox                45.82

[69 rows x 1 columns]




=============== ELECTRE III Flow ===============



                      Flow
ABB ADR              11.57
Accenture            51.66
SAP ADR               1.74
```

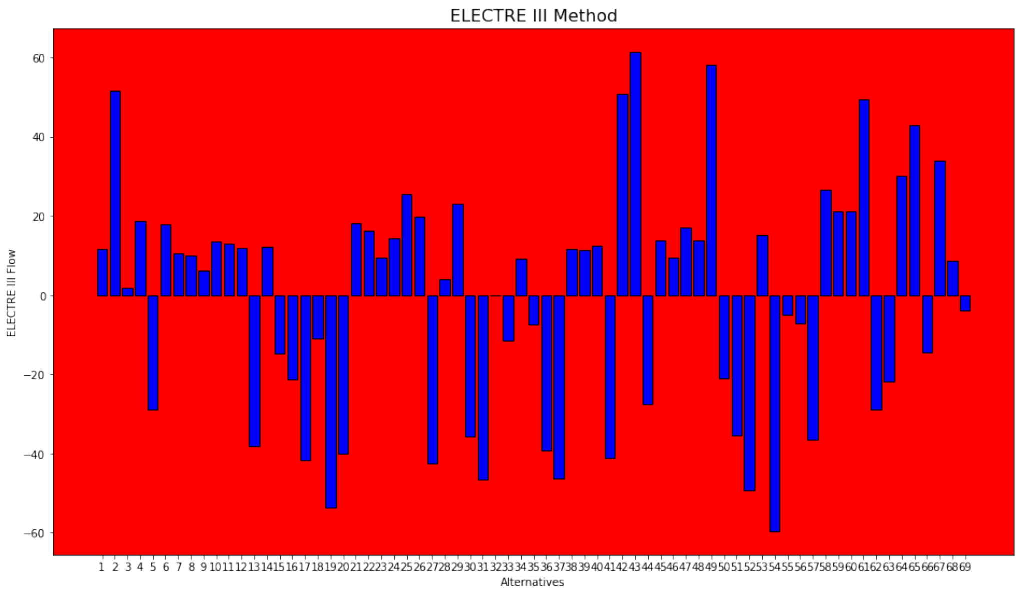
```
Infosys ADR            18.75
Wipro ADR             -28.89
...                      ...
Taiwan Semiconductor  42.89
Servotronics          -14.58
Synnex                 34.05
TransUnion              8.74
Xerox                  -3.92

[69 rows x 1 columns]
```

The final ranking of ELECTRE III method is provided if we sort the ELECTRE III flows. In the following cell we describe this procedure as well the visualisation of the results:

[16]:
```
tupledList = list(zip(companyName,phiEl))
tupledListSorted = sorted(tupledList, key=lambda tup: tup[1],␣
 ↪reverse=True)

print("=============== ELECTRE III Ranking ===============\n")
df = pd.DataFrame(tupledListSorted, columns=["Company", "Score"])
df.index = df.index + 1
display(df)

i = np.arange(alternatives)
plt.figure(figsize=(16,9))
plt.bar(i+1, phiEl, color = 'b', edgecolor = 'black')
plt.xlabel('Alternatives')
plt.ylabel('ELECTRE III Flow')
plt.title('ELECTRE III Method', fontsize=16)
ax = plt.gca()
ax.set_facecolor('red')
xlocs, xlabs = plt.xticks(np.arange(1, alternatives+1, step=1))
xlocs=[i for i in range(1,alternatives+1)]
xlabs=[i/2 for i in range(1,alternatives+1)]
# plt.xticks(xlocs, xlabs)


plt.show()
```

```
=============== ELECTRE III Ranking ===============


             Company  Score
```

```
1   Northrop Grumman  61.49
2                IBM  58.12
3          Accenture  51.66
4        GlobalSCAPE  50.81
5             Oracle  49.56
..               ...    ...
65           Enersys -46.35
66    DXC Technology -46.48
67             Kemet -49.29
68            Belden -53.64
69      Yirendai Ltd -59.48

[69 rows x 2 columns]
```



PROMETHEE II

Finally, the PROMETHEE method is also based on pairwise comparisons of the alternatives. It differs from ELECTRE III as it gives the opportunity to customise the comparison function based on the decision-maker's profile. Therefore, we must begin the presentation of this method with the different functions for pairwise comparisons.

[17]:
```
def usualCriterion(evaluationTable, k, alternatives,
    ↪decisionMatrix, indifferenceThreshold, preferenceThreshold,
    ↪weights, optimizationType):
```

```
    for i in range(alternatives):
        for j in range(alternatives):
            if i!=j and decisionMatrix[i][k] >=␣
↪decisionMatrix[j][k]:
                if decisionMatrix[i][k] > decisionMatrix[j][k]:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
↪+ 1.0 * weights[k]

def quasiCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType):
    for i in range(alternatives):
        for j in range(alternatives):
            if i!=j and decisionMatrix[i][k] >=␣
↪decisionMatrix[j][k]:
                if decisionMatrix[i][k] - decisionMatrix[j][k] >␣
↪indifferenceThreshold[k]:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
↪+ 1.0 * weights[k]

def linearPreferenceCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType):
    for i in range(alternatives):
        for j in range(alternatives):
            if i!=j and decisionMatrix[i][k] >=␣
↪decisionMatrix[j][k]:
                if decisionMatrix[i][k] - decisionMatrix[j][k] >␣
↪preferenceThreshold[k]:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
↪+ 1.0 * weights[k]
                else:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
↪+ ((decisionMatrix[i][k] - decisionMatrix [j][k])*1.0 /␣
↪preferenceThreshold[k]) * weights[k]

def levelCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType):
    for i in range(alternatives):
        for j in range(alternatives):
            if i!=j and decisionMatrix[i][k] >=␣
↪decisionMatrix[j][k]:
                if decisionMatrix[i][k] - decisionMatrix[j][k] >␣
↪preferenceThreshold[k]:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
↪+ 1.0 * weights[k]
```

```
                elif decisionMatrix[i][k] - decisionMatrix[j][k]␣
 ↪<= indifferenceThreshold[k]:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
 ↪+ 0.0 * weights[k]
                else:
                    evaluationTable[i][j] = evaluationTable[i][j]␣
 ↪+ 0.5 * weights[k]

def linearPreferenceAndIndifferenceCriterion(evaluationTable, k,␣
 ↪alternatives, decisionMatrix, indifferenceThreshold,␣
 ↪preferenceThreshold, weights, optimizationType):
    if optimizationType[k] == 0:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j and decisionMatrix[i][k] >=␣
 ↪decisionMatrix[j][k]:
                    if decisionMatrix[i][k] - decisionMatrix[j][k]␣
 ↪> preferenceThreshold[k]:
                        evaluationTable[i][j] =␣
 ↪evaluationTable[i][j] + 1.0 * weights[k]
                    elif decisionMatrix[i][k] -␣
 ↪decisionMatrix[j][k] > indifferenceThreshold[k]:
                        evaluationTable[i][j] =␣
 ↪evaluationTable[i][j] + ((decisionMatrix[i][k] - decisionMatrix␣
 ↪[j][k] - indifferenceThreshold[k])*1.0 /␣
 ↪(preferenceThreshold[k]-indifferenceThreshold[k])) * weights[k]
                    else:
                        evaluationTable[i][j] =␣
 ↪evaluationTable[i][j] + 0.0 * weights[k]
    elif optimizationType[k] == 1:
        for i in range(alternatives):
            for j in range(alternatives):
                if i!=j and decisionMatrix[i][k] >=␣
 ↪decisionMatrix[j][k]:
                    if decisionMatrix[i][k] - decisionMatrix[j][k]␣
 ↪> preferenceThreshold[k]:
                        evaluationTable[j][i] =␣
 ↪evaluationTable[j][i] + 1.0 * weights[k]
                    elif decisionMatrix[i][k] -␣
 ↪decisionMatrix[j][k] > indifferenceThreshold[k]:
                        evaluationTable[j][i] =␣
 ↪evaluationTable[j][i] + ((decisionMatrix[i][k] - decisionMatrix␣
 ↪[j][k] - indifferenceThreshold[k])*1.0 /␣
 ↪(preferenceThreshold[k]-indifferenceThreshold[k])) * weights[k]
                    else:
                        evaluationTable[j][i] =␣
 ↪evaluationTable[j][i] + 0.0 * weights[k]
```

After the definition of the necessary functions we can calculate the evaluation table between all the alternatives. The criterion used must be defined by the user in the .csv file:

[18]:
```
evaluationTable = [[0.0 for i in range(alternatives)] for y in␣
↪range(alternatives)]

for k in range(criteria):
    if criterion[k] == 1:
        usualCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType)
    elif criterion[k] == 2:
        quasiCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType)
    elif criterion[k] == 3:
        linearPreferenceCriterion(evaluationTable, k,␣
↪alternatives, decisionMatrix, indifferenceThreshold,␣
↪preferenceThreshold, weights, optimizationType)
    elif criterion[k] == 4:
        levelCriterion(evaluationTable, k, alternatives,␣
↪decisionMatrix, indifferenceThreshold, preferenceThreshold,␣
↪weights, optimizationType)
    elif criterion[k] == 5:
        linearPreferenceAndIndifferenceCriterion(evaluationTable,␣
↪k, alternatives, decisionMatrix, indifferenceThreshold,␣
↪preferenceThreshold, weights, optimizationType)
```

After the calculation of the evaluation table, we can compute the positive and negative flows just like in ELECTRE III method. The net flow of PROMETHEE method is the difference between these two flows:

[19]:
```
sumOfLines = np.sum(evaluationTable, axis=1)
sumOfColumns = np.sum(evaluationTable, axis=0)

phiPlus = sumOfLines*1.0 / (alternatives - 1)
phiMinus = sumOfColumns*1.0 / (alternatives - 1)
phi = phiPlus - phiMinus
phi2 = phiPlus - phiMinus

for i in range(alternatives):
    phi2[i] = round(round(phi[i],4) * 100,2)

print("============== Positive Flow ==============\n")
df = pd.DataFrame(phiPlus, index=companyName, columns=["Flow"])
```

```
display(df)
print("\n")
print("=============== Negative Flow ===============\n")
df = pd.DataFrame(phiMinus, index=companyName, columns=["Flow"])
display(df)
print("\n")
print("=============== PROMETHEE Net Flow ===============\n")
df = pd.DataFrame(phi, index=companyName, columns=["Flow"])
display(df)
```

```
=============== Positive Flow ===============



                          Flow
ABB ADR               0.188274
Accenture             0.438920
SAP ADR               0.255892
Infosys ADR           0.334441
Wipro ADR             0.102204
...                        ...
Taiwan Semiconductor  0.415894
Servotronics          0.194646
Synnex                0.497768
TransUnion            0.285903
Xerox                 0.319347

[69 rows x 1 columns]




=============== Negative Flow ===============



                          Flow
ABB ADR               0.411432
Accenture             0.058600
SAP ADR               0.293088
Infosys ADR           0.101991
Wipro ADR             0.418305
...                        ...
Taiwan Semiconductor  0.053415
Servotronics          0.363787
Synnex                0.048298
TransUnion            0.146850
```

```
Xerox                    0.279042

[69 rows x 1 columns]
```

```
=============== PROMETHEE Net Flow ===============
```

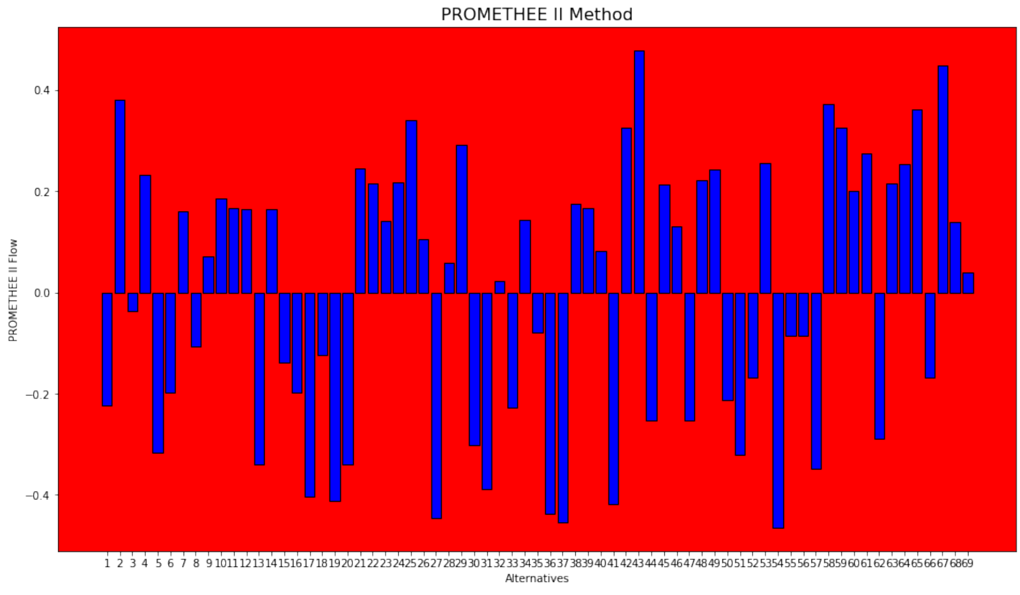
```
                             Flow
ABB ADR                 -0.223158
Accenture                0.380320
SAP ADR                 -0.037195
Infosys ADR              0.232450
Wipro ADR               -0.316101
...                           ...
Taiwan Semiconductor     0.362479
Servotronics            -0.169141
Synnex                   0.449470
TransUnion               0.139053
Xerox                    0.040305

[69 rows x 1 columns]
```

The final ranking of PROMETHEE II method is provided if we sort the net flows. In the following cell we describe this procedure as well the visualisation of the results:

[20]:
```
tupledList = list(zip(companyName,phiEl))
tupledListSorted = sorted(tupledList, key=lambda tup: tup[1],␣
 ↪reverse=True)

print("=============== PROMETHEE II Ranking ===============\n")
df = pd.DataFrame(tupledListSorted, columns=["Company", "Score"])
df.index = df.index + 1
display(df)

i = np.arange(alternatives)
plt.figure(figsize=(16,9))
plt.bar(i+1, phi, color = 'b', edgecolor = 'black')
plt.xlabel('Alternatives')
plt.ylabel('PROMETHEE II Flow')
plt.title('PROMETHEE II Method', fontsize=16)
ax = plt.gca()
```

```
ax.set_facecolor('red')
xlocs, xlabs = plt.xticks(np.arange(1, alternatives+1, step=1))
xlocs=[i for i in range(1,alternatives+1)]
xlabs=[i/2 for i in range(1,alternatives+1)]
# plt.xticks(xlocs, xlabs)


plt.show()
```

```
=============== PROMETHEE II Ranking ===============


              Company  Score
1   Northrop Grumman  61.49
2                IBM  58.12
3          Accenture  51.66
4        GlobalSCAPE  50.81
5             Oracle  49.56
..               ...    ...
65           Enersys -46.35
66    DXC Technology -46.48
67             Kemet -49.29
68           Belden -53.64
69      Yirendai Ltd -59.48

[69 rows x 2 columns]
```



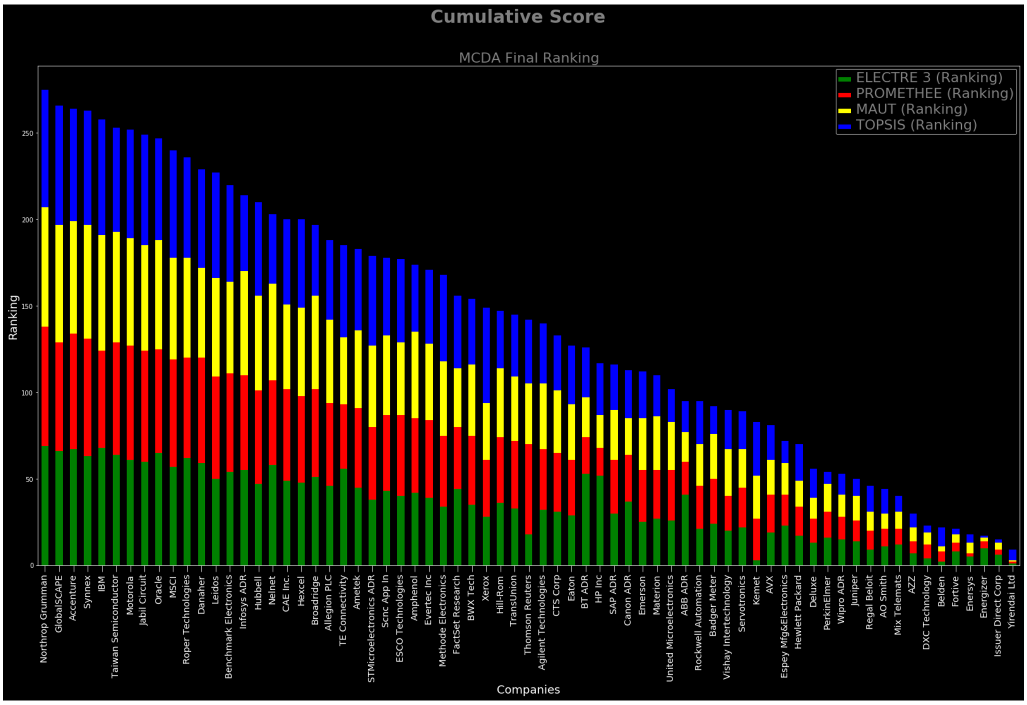
Cumulative Ranking

The final ranking is the weighted sum of the rankings of the four MCDA methods. Therefore, in the following cell we describe the calculation of the cumulative score of each alternative as well as the final ranking and a visualisation of the results.

[21]:
```
decisionMatrix = np.array(decisionMatrix)
data = {'Name':companyName, 'P/E Ratio':decisionMatrix[:,0],'EPS':
 ↪decisionMatrix[:,1],'Revenue (B)':decisionMatrix[:,2],
            'Beta':decisionMatrix[:,3],'Dividend Yield':
 ↪decisionMatrix[:,4],'Monthly':decisionMatrix[:,5],'YTD':
 ↪decisionMatrix[:,6],
            '1Year':decisionMatrix[:,7], 'ELECTRE 3':phiEl, 'MAUT':
 ↪utilityScorePer, 'PROMETHEE':phi, 'TOPSIS':C2}

df = pd.DataFrame(data)
df["ELECTRE 3 (Ranking)"] = df["ELECTRE 3"].rank()
df["MAUT (Ranking)"] = df["MAUT"].rank()
df["PROMETHEE (Ranking)"] = df["PROMETHEE"].rank()
df["TOPSIS (Ranking)"] = df["TOPSIS"].rank()

df["RankSum"] = df.iloc[:,13:17].sum(axis=1)
df.sort_values("RankSum", inplace=True, ascending=False)
display(df)

plt.style.use('dark_background')
ax = df.plot(x ='Name',
                y={'ELECTRE 3 (Ranking)','MAUT (Ranking)','TOPSIS␣
 ↪(Ranking)','PROMETHEE (Ranking)'},
            color={'green','red','yellow','blue'},
            kind = 'bar',
            figsize=(24, 18),
            stacked=True,
            rot=90)
plt.suptitle("Cumulative Score", color='grey', size='28',␣
 ↪fontweight='bold')
ax.set_title('MCDA Final Ranking', color='grey', size='22')
ax.tick_params(axis='x', colors='white', length=10,␣
 ↪labelsize='x-large')
ax.tick_params(axis='y', colors='white')
legend = plt.legend()
plt.setp(legend.get_texts(), color='grey', size='22')
plt.xlabel("Companies", color='white', size='18')
plt.ylabel("Ranking", color='white', size='18')
plt.subplots_adjust(left=0.06, bottom=0.3, right=0.96, top=0.91,␣
 ↪wspace=0.2, hspace=0.2)
```

```
                 Name  P/E Ratio    EPS  Revenue (B)  Beta  Dividend Yield  \
42   Northrop Grumman      18.15  20.26       32.890  0.80            1.44
41        GlobalSCAPE      21.23   0.55        0.037  0.57            0.52
1           Accenture      25.05   7.36       43.220  1.05            1.74
66             Synnex      12.63   8.91       22.800  0.99            1.33
48                IBM      11.75  12.01       77.860  1.36            4.59
..                ...        ...    ...          ...   ...             ...
40            Fortive      36.69   1.77        6.310  1.22            0.43
36            Enersys      17.55   3.47        2.920  1.60            1.15
35          Energizer      91.94   0.43        2.230  0.67            3.05
26  Issuer Direct Corp     61.59   0.17        0.015  0.76            1.95
53        Yirendai Ltd      3.17   2.07        2.550  2.70            4.27

    Monthly     YTD   1Year  ELECTRE 3   MAUT  PROMETHEE  TOPSIS  \
42      5.0   50.57   21.95      61.49  69.84   0.478513   50.88
41      5.0  174.90  204.11      50.81  67.71   0.326386   62.82
1       5.0   30.37   15.30      51.66  62.42   0.380320   43.95
66      5.0   39.26   51.26      34.05  63.04   0.449470   47.58
48      4.0   24.16    0.20      58.12  63.81   0.243433   48.62
..      ...     ...     ...        ...    ...        ...     ...
40      1.0   -3.30  -17.39     -41.10  19.16  -0.418051   25.60
36      1.0  -21.48  -23.75     -46.35  19.29  -0.454859   26.38
35      1.0  -12.76  -33.70     -39.19  17.75  -0.436301   19.48
26      1.0   -9.43  -31.92     -42.43  18.35  -0.445197   21.92
53      1.0  -38.98  -59.25     -59.48  16.64  -0.463767   26.48

    ELECTRE 3 (Ranking)  MAUT (Ranking)  PROMETHEE (Ranking)  \
42                 69.0            69.0                 69.0
41                 66.0            68.0                 63.0
1                  67.0            65.0                 67.0
66                 63.0            66.0                 68.0
48                 68.0            67.0                 56.0
..                  ...             ...                  ...
40                  8.0             5.0                  5.0
36                  5.0             6.0                  2.0
35                 10.0             2.0                  4.0
26                  6.0             4.0                  3.0
53                  1.0             1.0                  1.0

    TOPSIS (Ranking)  RankSum
42              68.0    275.0
41              69.0    266.0
1               65.0    264.0
66              66.0    263.0
48              67.0    258.0
..               ...      ...
40               3.0     21.0
36               5.0     18.0
```

```
35                 1.0     17.0
26                 2.0     15.0
53                 6.0      9.0

[69 rows x 18 columns]
```



Phase II: Multiobjective Portfolio Optimisation

Goal Programming methodology

[1]:
```
from mip import *
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from pandas_datareader import data
```

```
Using Python-MIP package version 1.6.2
```

Input

The input of this method includes the following variables:

- numSecurities (integer) is the number of securities,
- securityName (str list) contains the names of the securities,
- betaIndex (float list) contains the beta index for each security,
- DYIndex (float list) contains the dividend yield for each security.

[2]:
```

numSecurities = 10

tickers = ['AAPL', 'MSFT', 'AUDC', 'KLAC', 'TTEC', 'SLP', 'EQIX',␣
 ↪'WSTG', 'CDW', 'TER']

betaIndex = [1.22, 1.23, 0.56, 1.71, 0.7, -0.32, 0.7, 0.39, 1.09,␣
 ↪1.56]

Rev = [259.03, 125.84, 0.186, 4.57, 1.57, 0.032, 5.34, 0.192, 17.
 ↪04, 2.14]

DYIndex = [1.36, 1.48, 0.76, 1.91, 1.38, 0.72, 1.71, 4.63, 0.97, 0.
 ↪6]
```

Model Parameters

The parameters of the model are defined below:

- numPortfolios : Number of portfolios to be constructed.

- minSecurities : Minimum number of securities to participate in each portfolio.
- maxSecurities : Maximum number of securities to participate in each portfolio.
- lowerBound : Minimum value of the weight of each security.
- upperBound : Maximum value of the weight of each security.
- capitalThreshold : Threshold that determines the Billions needed to consider a security as a high capitalisation investment

The target values of the goal programming model are set as follows:

- betaGoal : The target value for portfolio beta
- DYGoal : The target value for portfolio Dividend Yield
- highCapGoal : The target value for the percentage of high capitalisation securities participating in the portfolio

[3]:
```
minSecurities = 4
maxSecurities = 10
lowerBound = 0.1
upperBound = 0.4
betaGoal = 0.7
DYGoal = 1.7
highCapGoal = 0.5
capitalThreshold = 100
```

The model is constructed below:

[4]:
```
m = Model()

high = [0 for i in range(numSecurities)]
for i in range(numSecurities):
    if Rev[i] > capitalThreshold:
        high[i] =1
print(high)

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
 ↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
 ↪range(numSecurities) ]

d1P = m.add_var(var_type=CONTINUOUS)
d1M = m.add_var(var_type=CONTINUOUS)
d2P = m.add_var(var_type=CONTINUOUS)
d2M = m.add_var(var_type=CONTINUOUS)
d3P = m.add_var(var_type=CONTINUOUS)
d3M = m.add_var(var_type=CONTINUOUS)

w1P = 1
```

```
w1M = 1
w2P = 1
w2M = 1
w3P = 1
w3M = 1

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m += xsum(weights[i] * betaIndex[i] for i in range(numSecurities)) + d1M - d1P == betaGoal
m += xsum(weights[i] * DYIndex[i] for i in range(numSecurities)) + d2M - d2P == DYGoal
m += xsum(weights[i] * high[i] for i in range(numSecurities)) + d3M - d3P == highCapGoal

m.objective = minimize((w1P * d1P + w1M * d1M) / betaGoal + (w2P * d2P + w2M * d2M) / DYGoal + (w3P * d3P + w3M * d3M) / highCapGoal)

status = m.optimize()

print("=================  Model output  ================= ")

print("Solution status : ", status, "\n")
obj = m.objective_value
print("Objective function = ", obj, "\n")


for i in range(numSecurities):
    print(tickers[i],": ", weights[i].x * onoff[i].x)
print()


print("Portfolio beta = ", sum(weights[i].x * betaIndex[i] for i in range(numSecurities)))
print("Portfolio dividend yield = ", sum(weights[i].x * DYIndex[i] for i in range(numSecurities)))
print("High capitalisation percentage", sum(weights[i].x * high[i] for i in range(numSecurities)))

portfolio = [0 for i in range(numSecurities)]
for i in range(numSecurities):
```

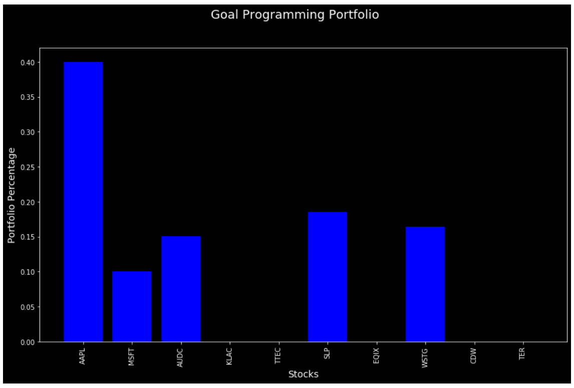
```
    portfolio[i] = onoff[i].x * weights[i].x
```

```
=================  Model output  =================
Solution status :  OptimizationStatus.OPTIMAL

Objective function =  0.0

AAPL :  0.4
MSFT :  0.1
AUDC :  0.1504835306529128
KLAC :  0.0
TTEC :  0.0
SLP :  0.18532704255069749
EQIX :  0.0
WSTG :  0.1641894267963896
CDW :  0.0
TER :  0.0

Portfolio beta =  0.7
Portfolio dividend yield =  1.6999999999999997
High capitalisation percentage 0.5
```

Finally, the following cell contains the code of the portfolio visualisation.

[5]:
```
from pandas.plotting import register_matplotlib_converters
register_matplotlib_converters()

plt.style.use('dark_background')
fig = plt.figure(figsize=(14, 8))
plt.bar(tickers,portfolio, color='b')
plt.suptitle('Goal Programming Portfolio', color='white',␣
 ↪size='18')
plt.xticks(tickers, tickers, rotation='vertical')
plt.xlabel('Stocks', size='14')
plt.ylabel('Portfolio Percentage', size='14')
```

[5]:



Genetic algorithm model

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import openpyxl
from scipy.optimize import differential_evolution
from scipy.optimize import LinearConstraint, minimize
from pandas_datareader import data
```

Securities Input Data

The input data of this model are the namesand the values of the securities. The values are imported from yahoo finance database for a selected time period, which is determined with the variables startDate and endDate. The nesxt step involves the calculation of the arithmetic return for the selected securities.

[2]:
```
#NASDAQ
tickers = ['AAPL', 'MSFT', 'AUDC', 'KLAC', 'TTEC', 'SLP', 'EQIX',␣
 ↪'WSTG', 'CDW', 'TER']

startDate = '2016-01-01'
endDate = '2018-12-31'

historicalValues = data.DataReader(tickers, 'yahoo', startDate,␣
 ↪endDate)
stockValues = historicalValues['Open']
numOfDates = stockValues.shape[0]
numSecurities = stockValues.shape[1]
stockValues = stockValues.fillna(method='ffill')

numPeriods = numOfDates - 1
stockValuesArray = pd.DataFrame(stockValues).to_numpy()
secReturns = np.empty(shape = (numPeriods, numSecurities))
for i in range(numSecurities):
    for j in range(numPeriods):
        secReturns[j][i] =␣
 ↪(stockValuesArray[j+1][i]-stockValuesArray[j][i])/
 ↪stockValuesArray[j][i]
returnDates = stockValues.index[1:]
```

```
stockReturns = pd.DataFrame(secReturns, index=returnDates,␣
↪columns=stockValues.columns)
print("============  Stock Returns  ============ \n")
display(stockReturns)
```

```
============  Stock Returns  ============


Symbols          AAPL      MSFT      AUDC      KLAC      TTEC       SLP  \
Date
2016-01-05  0.030601  0.011230  0.086842  0.005145 -0.026536 -0.015448
2016-01-06 -0.049078 -0.011105 -0.041162 -0.009798 -0.037715  0.014644
2016-01-07 -0.018695 -0.029823  0.000000 -0.033821 -0.008537  0.019588
2016-01-08 -0.001317 -0.006262  0.032828  0.011006  0.001174  0.006067
2016-01-11  0.004262  0.002673 -0.007335 -0.005745 -0.008210  0.030151
...              ...       ...       ...       ...       ...       ...
2018-12-24 -0.055527 -0.038866 -0.064453 -0.025800 -0.045767 -0.002847
2018-12-26  0.001013 -0.026003  0.003131 -0.029912 -0.005196  0.006282
2018-12-27  0.050843  0.043725  0.002081  0.032663  0.054640  0.036890
2018-12-28  0.010652  0.028097  0.008307  0.042134  0.027429  0.042146
2018-12-31  0.006540 -0.007836  0.020597  0.019366  0.024101 -0.027836

Symbols          EQIX      WSTG       CDW       TER
Date
2016-01-05 -0.005905  0.009387 -0.007969 -0.005865
2016-01-06  0.005571 -0.007659 -0.017770 -0.010325
2016-01-07  0.016987  0.019846  0.002726 -0.032290
2016-01-08 -0.000427  0.000000 -0.009145  0.005133
2016-01-11  0.023441  0.027027 -0.008481 -0.014811
...              ...       ...       ...       ...
2018-12-24 -0.027522 -0.044118 -0.044309 -0.027833
2018-12-26 -0.023943  0.021538 -0.015584  0.000682
2018-12-27  0.015423 -0.031125  0.035488  0.016349
2018-12-28  0.008969  0.006218  0.021659  0.066354
2018-12-31  0.010323  0.021627  0.003866 -0.006600

[753 rows x 10 columns]
```

Market Index Input Data

The same process is followed for the market index. In this case we select the NASDAQ Composite index (ˆIXIC) because the seleted securities belong to NASDAQ stock exchange. Therefore, we calculate the returns of the market index for the selected time horizon.

[3]:
```
historicalValues = data.DataReader('^IXIC', 'yahoo', startDate,␣
↪endDate)
indexValues = historicalValues['Open']
numOfDates = stockValues.shape[0]
indexValues = indexValues.fillna(method='ffill')
display(indexValues)

numPeriods = numOfDates - 1
indexValuesArray = pd.DataFrame(indexValues).to_numpy()
marketReturns = np.empty(shape = (numPeriods))
for j in range(numPeriods):
    marketReturns[j] = (indexValuesArray[j+1]-indexValuesArray[j])/
↪indexValuesArray[j]
indexReturns = pd.DataFrame(marketReturns, index=returnDates)
print("============  Stock Returns  ============ \n")
display(indexReturns)
```

```
============  Stock Returns  ============



                   0
Date
2016-01-05  0.004122
2016-01-06 -0.021164
2016-01-07 -0.016071
2016-01-08 -0.003036
2016-01-11 -0.010288
...              ...
2018-12-24 -0.044877
2018-12-26 -0.003286
2018-12-27  0.031853
2018-12-28  0.024717
2018-12-31  0.004947

[753 rows x 1 columns]
```

Multiobjective Function

In this point, we define the multiobjective function losses, which is about to be optimised. This function calculates the number of times that the portfolio return is smaller than the selected marfket index return for all dates during the selected time period. This function needs to be minimised, as the target is to maximise the number of times that the portfolio offers a better return.

[4]:
```
def losses(x):
    losingTimes = 0
    portfReturn = [0 for i in range(numPeriods)]
    for i in range(numPeriods):
        for j in range(numSecurities):
            portfReturn[i] += x[j] * secReturns[i][j]
        if portfReturn[i] < marketReturns[i]:
            losingTimes += 1
    return losingTimes
```

Finally, the function differential_evolution solves this evolutionary problem. The list of variables bounds sets the limits for each security's percentage in the portfolio. The function LinearConstraint imposes the capital completeness constraint. The parameters of the model are the following (according to the scipy documentation):

- maxiter: The maximum number of generations over which the entire population is evolved.
- popsizeint: A multiplier for setting the total population size. The population has popsize * len(x) individuals.
- mutation: The mutation constant. In the literature this is also known as differential weight, being denoted by F.
- recombination: The recombination constant, should be in the range [0, 1]. Also known as the crossover probability, being denoted by CR. Increasing this value allows a larger number of mutants to progress into the next generation, but at the risk of population stability.

[5]:
```
bounds = [(0,1) for i in range(numSecurities)]

constraints = LinearConstraint(np.ones(numSecurities), 1, 1)
result = differential_evolution(losses, bounds, maxiter=5000,␣
 ↪polish=True, popsize=10, mutation=1.95,
                                updating='deferred',␣
 ↪recombination=0.05, constraints=constraints)

for i in range(numSecurities):
    print(tickers[i],": ", np.round(result.x[i],4))
print("Percentage that portfolio beats the index:",␣
 ↪(numPeriods-result.fun)/numPeriods)
```

```
AAPL :  0.1226
MSFT :  0.1568
AUDC :  0.0018
KLAC :  0.0684
TTEC :  0.1567
```

```
SLP :  0.0
EQIX :  0.0668
WSTG :  0.1488
CDW :  0.1578
TER :  0.1202
Percentage of portfolio wins over the market index: 0.6099
```

MOIP PROMETHEE flow 2-Dimensional model

[1]:
```
from mip import *
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from pandas_datareader import data
```

```
Using Python-MIP package version 1.6.2
```

Input

The input of this method includes the following variables:

- numSecurities (integer) is the number of securities,
- tickers (str list) contains the names of the securities,
- betaIndex (float list) contains the beta index for each security,
- promIndex (float list) contains the dividend yield for each security.

[2]:
```
numSecurities = 10

tickers = ['AAPL', 'MSFT', 'AUDC', 'KLAC', 'TTEC', 'SLP', 'EQIX',␣
↪'WSTG', 'CDW', 'TER']

betaIndex = [1.22, 1.23, 0.56, 1.71, 0.7, -0.32, 0.7, 0.39, 1.09,␣
↪1.56]

promIndex = [0.3136, 0.2861, 0.2999, 0.2978, 0.2840, 0.2683, 0.
↪2728, 0.3078, 0.2552, 0.1867]
```

Model Parameters

The model contains the following parameters:

- numPortfolios : Number of portfolios to be constructed.
- minSecurities : Minimum number of securities to participate in each portfolio.
- maxSecurities : Maximum number of securities to participate in each portfolio.
- lowerBound : Minimum value of the weight of each security.
- upperBound : Maximum value of the weight of each security.

[3]:
```
numPortfolios = 20
minSecurities = 4
maxSecurities = 10
lowerBound = 0.05
upperBound = 0.5
```

Minimum Beta Portfolio

The first step of the process involves finding the portfolio which has the minimum portfolio beta:

[4]:
```
m = Model()

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
↪range(numSecurities) ]

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m.objective = minimize(xsum(weights[i] * betaIndex[i] for i in␣
↪range(numSecurities)))

status = m.optimize()

print(status, "\n")
minBeta = m.objective_value
print("Minimum Beta = ", minBeta, "\n")
```

```
OptimizationStatus.OPTIMAL

Minimum Beta =  0.059
```

Maximum Beta Portfolio

The next step of the process involves finding the portfolio which has the maximum portfolio beta:

[5]:
```
m = Model()

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
 ↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
 ↪range(numSecurities) ]

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m.objective = maximize(xsum(weights[i] * betaIndex[i] for i in␣
 ↪range(numSecurities)))

status = m.optimize()

print(status, "\n")
maxBeta = m.objective_value
print("Maximum Beta = ", maxBeta, "\n")
```

```
OptimizationStatus.OPTIMAL

Maximum Beta =  1.6015000000000001
```

Multiobjective Integer Programming Model

The problem is solved parametrically for various portfolio beta indexes between the minimum and maximum portfolio beta calculated in the previous steps. Therefore, in the following cell the procedure of finding the efficient frontier is described. The model is equipped with the following variables:

- m represents an empty MILP model with default settings.
- onoff is a list of binary variables.
- weights is a list with continuous variables representing the weighting factor of each security.

- portfolios is a 2-dimensional list where the pareto optimal portfolios are saved
- netFlows and betas are two lists which contain the PROMETHEE net flow and the beta index for each portfolio.

[6]:
```
portfolios = [[0 for i in range(numSecurities)] for j in␣
↪range(numPortfolios)]
netFlows = [0 for i in range(numPortfolios)]
betas = [0 for i in range(numPortfolios)]

betas = np.linspace(minBeta, maxBeta, num=numPortfolios)
print(betas)



for k in range(numPortfolios):
    portfolioBeta = betas[k]

    m = Model()

    onoff = [ m.add_var(var_type=BINARY) for i in␣
↪range(numSecurities) ]
    weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
↪range(numSecurities) ]

    m += xsum(weights[i] for i in range(numSecurities)) == 1

    m += xsum(onoff[i] for i in range(numSecurities)) <=␣
↪maxSecurities
    m += xsum(onoff[i] for i in range(numSecurities)) >=␣
↪minSecurities

    for i in range(numSecurities):
        m += weights[i] - lowerBound * onoff[i] >= 0
        m += weights[i] - upperBound * onoff[i] <= 0

    m += xsum(weights[i] * betaIndex[i] for i in␣
↪range(numSecurities)) == portfolioBeta

    m.objective = maximize(xsum(weights[i] * promIndex[i] for i in␣
↪range(numSecurities)))

    status = m.optimize()

    netFlows[k] = m.objective_value
    for i in range(numSecurities):
        portfolios[k][i] = onoff[i].x * weights[i].x
```

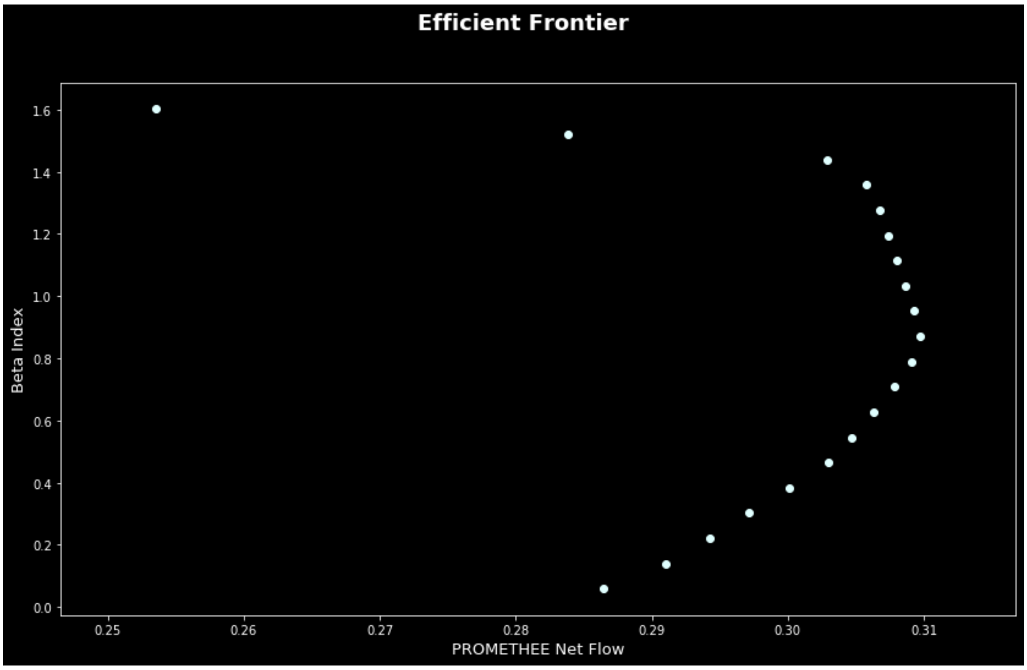
```
for i in range(numPortfolios):
    print("Portfolio", i, ":  Beta = ", betas[i], " Net Flow = ",␣
↪netFlows[i] )
```

```
Portfolio 0 :  Beta =  0.059  Net Flow =  0.286465
Portfolio 1 :  Beta =  0.1401842105263158  Net Flow =  0.
↪2910151074870274
Portfolio 2 :  Beta =  0.2213684210526316  Net Flow =  0.
↪2942423205741627
Portfolio 3 :  Beta =  0.3025526315789474  Net Flow =  0.
↪29715757177033497
Portfolio 4 :  Beta =  0.3837368421052632  Net Flow =  0.
↪30007282296650717
Portfolio 5 :  Beta =  0.464921052631579  Net Flow =  0.
↪30297411881977676
Portfolio 6 :  Beta =  0.5461052631578949  Net Flow =  0.
↪30465930622009574
Portfolio 7 :  Beta =  0.6272894736842105  Net Flow =  0.
↪3063444936204146
Portfolio 8 :  Beta =  0.7084736842105264  Net Flow =  0.
↪30784415345592897
Portfolio 9 :  Beta =  0.7896578947368422  Net Flow =  0.
↪3090829744816587
Portfolio 10 :  Beta =  0.8708421052631581  Net Flow =  0.
↪3097444990488269
Portfolio 11 :  Beta =  0.9520263157894737  Net Flow =  0.
↪30925555821371614
Portfolio 12 :  Beta =  1.0332105263157896  Net Flow =  0.
↪30864052631578953
Portfolio 13 :  Beta =  1.1143947368421054  Net Flow =  0.
↪30802549441786287
Portfolio 14 :  Beta =  1.195578947368421  Net Flow =  0.
↪3074104625199362
Portfolio 15 :  Beta =  1.276763157894737  Net Flow =  0.
↪3067954306220096
Portfolio 16 :  Beta =  1.3579473684210528  Net Flow =  0.
↪30577465628356604
Portfolio 17 :  Beta =  1.4391315789473687  Net Flow =  0.
↪3028568796992481
Portfolio 18 :  Beta =  1.5203157894736845  Net Flow =  0.
↪2838658126934983
Portfolio 19 :  Beta =  1.6015000000000001  Net Flow =  0.
↪25356499999999993
```

The efficient frontier is visualised below:

[7]:
```
from pandas.plotting import register_matplotlib_converters
register_matplotlib_converters()

plt.style.use('dark_background')

fig, ax = plt.subplots(figsize=(14,8))
ax.scatter(netFlows, betas, color='lightcyan')
plt.suptitle('Efficient Frontier', fontsize=18, fontweight='bold')
ax.set_xlabel('PROMETHEE Net Flow', fontsize=13)
ax.set_ylabel('Beta Index', fontsize=13)
```

[7]:



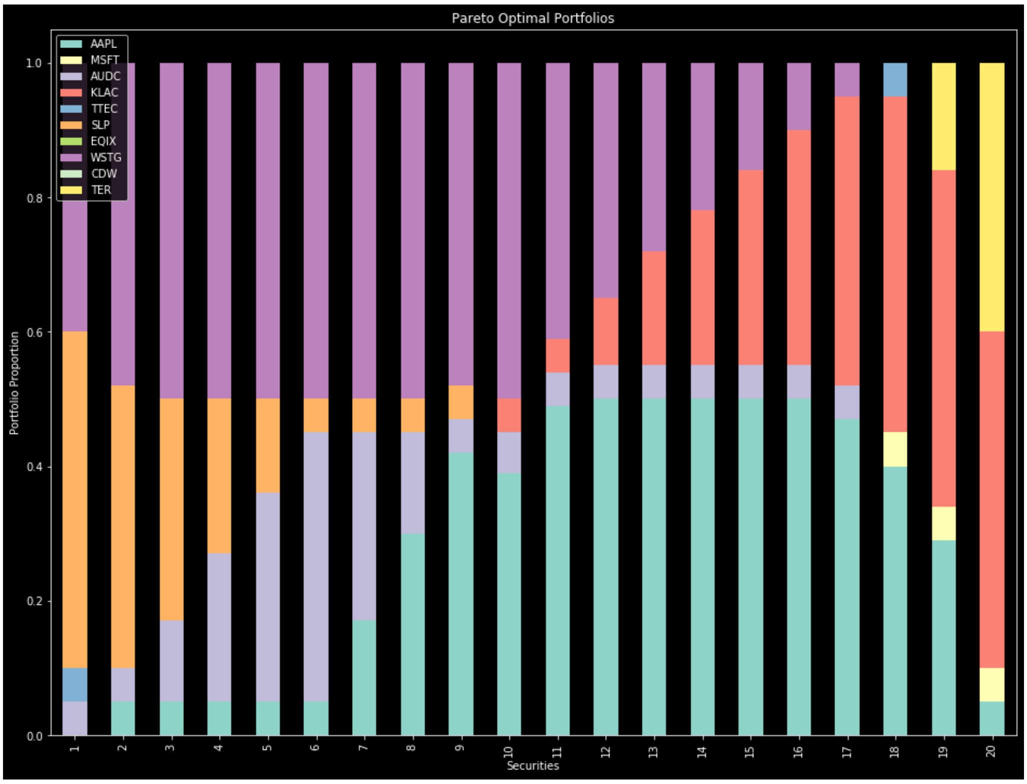
Finally, the constructed portfolios are presented in the following table:

[8]:
```
df = pd.DataFrame(np.round(portfolios,2))
df.index = df.index + 1
df.columns = tickers
display(df)
ax = df.plot.bar(stacked=True, figsize=(16, 12), title ='Pareto␣
 ↪Optimal Portfolios')
ax.set_xlabel("Securities")
ax.set_ylabel("Portfolio Proportion")
```

```
    AAPL  MSFT  AUDC  KLAC  TTEC   SLP  EQIX  WSTG  CDW   TER
```

1	0.00	0.00	0.05	0.00	0.05	0.50	0.0	0.40	0.0	0.00
2	0.05	0.00	0.05	0.00	0.00	0.42	0.0	0.48	0.0	0.00
3	0.05	0.00	0.12	0.00	0.00	0.33	0.0	0.50	0.0	0.00
4	0.05	0.00	0.22	0.00	0.00	0.23	0.0	0.50	0.0	0.00
5	0.05	0.00	0.31	0.00	0.00	0.14	0.0	0.50	0.0	0.00
6	0.05	0.00	0.40	0.00	0.00	0.05	0.0	0.50	0.0	0.00
7	0.17	0.00	0.28	0.00	0.00	0.05	0.0	0.50	0.0	0.00
8	0.30	0.00	0.15	0.00	0.00	0.05	0.0	0.50	0.0	0.00
9	0.42	0.00	0.05	0.00	0.00	0.05	0.0	0.48	0.0	0.00
10	0.39	0.00	0.06	0.05	0.00	0.00	0.0	0.50	0.0	0.00
11	0.49	0.00	0.05	0.05	0.00	0.00	0.0	0.41	0.0	0.00
12	0.50	0.00	0.05	0.10	0.00	0.00	0.0	0.35	0.0	0.00
13	0.50	0.00	0.05	0.17	0.00	0.00	0.0	0.28	0.0	0.00
14	0.50	0.00	0.05	0.23	0.00	0.00	0.0	0.22	0.0	0.00
15	0.50	0.00	0.05	0.29	0.00	0.00	0.0	0.16	0.0	0.00
16	0.50	0.00	0.05	0.35	0.00	0.00	0.0	0.10	0.0	0.00
17	0.47	0.00	0.05	0.43	0.00	0.00	0.0	0.05	0.0	0.00
18	0.40	0.05	0.00	0.50	0.05	0.00	0.0	0.00	0.0	0.00
19	0.29	0.05	0.00	0.50	0.00	0.00	0.0	0.00	0.0	0.16
20	0.05	0.05	0.00	0.50	0.00	0.00	0.0	0.00	0.0	0.40

[8]:



MOIP PROMETHEE flow 3-Dimensional Model

In this paragraph there is a presentation of the 3-Dimensional model involving the PROMETHEE flow. The 3 objective function will be the portfolio beta index, the portfolio dividend yield and the PROMETHEE flow. The input data for this method are the beta indexes, the PROMETHEE net flows and the dividend yield of each security:

[1]:
```
from mip import *
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

numSecurities = 10

tickers = ['AAPL', 'MSFT', 'AUDC', 'KLAC', 'TTEC', 'SLP', 'EQIX',␣
↪'WSTG', 'CDW', 'TER']

betaIndex = [1.22, 1.23, 0.56, 1.71, 0.7, -0.32, 0.7, 0.39, 1.09,␣
↪1.56]

promIndex = [0.3136, 0.2861, 0.2999, 0.2978, 0.2840, 0.2684, 0.
↪2728, 0.3078, 0.2552, 0.1867, 0.2351]

DYIndex = [1.36, 1.48, 0.76, 1.91, 1.38, 0.72, 1.71, 4.63, 0.97, 0.
↪6]
```

```
Using Python-MIP package version 1.6.2
```

Model Parameters

The model contains the following parameters:

- minSecurities : Minimum number of securities to participate in each portfolio.
- maxSecurities : Maximum number of securities to participate in each portfolio.
- lowerBound : Minimum value of the weight of each security.
- upperBound : Maximum value of the weight of each security.

[2]:
```
minSecurities = 6
maxSecurities = 10
lowerBound = 0.05
upperBound = 0.3
```

Determination of the objectives target values

We solve the 1-objective optimisation problem for each one of the objective functions, in order to find their target values. Firstly, we solve the problem of minimising the portfolio beta.

[3]:
```
m = Model()

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
↪range(numSecurities) ]

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m.objective = minimize(xsum(weights[i] * betaIndex[i] for i in␣
↪range(numSecurities)))

status = m.optimize()

print(status, "\n")
minBeta = m.objective_value
print("Minimum Beta = ", minBeta, "\n")
```

```
OptimizationStatus.OPTIMAL

Minimum Beta =  0.2854999999999999
```

Secondly, we solve the maximisation problem of the PROMETHEE flow function.

[4]:
```
m = Model()

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
↪range(numSecurities) ]

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
```

```
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m.objective = maximize(xsum(weights[i] * promIndex[i] for i in␣
 ↪range(numSecurities)))

status = m.optimize()

print(status, "\n")
maxProm = m.objective_value
print("Maximum PROMETHEE flow = ", maxProm, "\n")
```

```
OptimizationStatus.OPTIMAL

Maximum PROMETHEE flow =  0.30479
```

Finally, we solve the problem of maximising the portfolio dividend yield.

[5]:
```
m = Model()

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
 ↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
 ↪range(numSecurities) ]

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m.objective = maximize(xsum(weights[i] * DYIndex[i] for i in␣
 ↪range(numSecurities)))

status = m.optimize()

print(status, "\n")
maxDY = m.objective_value
print("Maximum DY = ", maxDY, "\n")
```

```
OptimizationStatus.OPTIMAL
```

```
Maximum DY =  2.6005
```

Minimax objective Optimisation Problem

In the following cell we set the final problem as a goal programming optimisation problem with the minimax objective. The results for a random selection of offsets is presented below:

[6]:
```
m = Model()

w1 = 0.2
w2 = 0.1
w3 = 0.7

onoff = [ m.add_var(var_type=BINARY) for i in range(numSecurities)␣
 ↪]
weights = [ m.add_var(var_type=CONTINUOUS) for i in␣
 ↪range(numSecurities) ]
Q = m.add_var(var_type=CONTINUOUS)

m += xsum(weights[i] for i in range(numSecurities)) == 1
m += xsum(onoff[i] for i in range(numSecurities)) <= maxSecurities
m += xsum(onoff[i] for i in range(numSecurities)) >= minSecurities
for i in range(numSecurities):
    m += weights[i] - lowerBound * onoff[i] >= 0
    m += weights[i] - upperBound * onoff[i] <= 0

m += w1 * ((xsum(weights[i] * betaIndex[i] for i in␣
 ↪range(numSecurities))) - minBeta) / minBeta <= Q
m += w2 * (maxProm - (xsum(weights[i] * promIndex[i] for i in␣
 ↪range(numSecurities)))) / maxProm <= Q
m += w3 * (maxDY - (xsum(weights[i] * DYIndex[i] for i in␣
 ↪range(numSecurities)))) / maxDY <= Q


m.objective = minimize(Q)

status = m.optimize()

print(status, "\n")
minQ = m.objective_value
print("Q = ", minQ, "\n")

for i in range(numSecurities):
    print(tickers[i],": ", weights[i].x * onoff[i].x)
```

```
print()

print("Portfolio beta = ", sum(weights[i].x * betaIndex[i] for i␣
↪in range(numSecurities)))
print("Portfolio PROMETHEE flow = ", sum(weights[i].x *␣
↪promIndex[i] for i in range(numSecurities)))
print("Portfolio DY = ", sum(weights[i].x * DYIndex[i] for i in␣
↪range(numSecurities)))
```

```
OptimizationStatus.OPTIMAL

Q =  0.08220335317946557

AAPL :  0.0
MSFT :  0.05
AUDC :  0.0
KLAC :  0.049999999999999996
TTEC :  0.05
SLP :  0.27564187581991456
EQIX :  0.27435812418008565
WSTG :  0.29999999999999993
CDW :  0.0
TER :  0.0

Portfolio beta =  0.4028452866636873
Portfolio PROMETHEE flow =  0.2845621757463924
Portfolio DY =  2.2951145429382844
```

Appendix B

Extensive Experimental Results

This section covers the presentation of the empirical testing results for the stock exchanges of NASDAQ, Paris and Tokyo. The objective of this section is to provide a series of tables which describe the application of the methodology step-by-step for each stock exchange. The structure of this chapter is according to the structure of the NYSE stock exchange results presentation in chapter 6. More specifically, the chapter is divided into three sections:

1. **NASDAQ stock exchange**: In tables B.1 - B.3 the input matrix for the three industrial segments is presented. Tables B.4 - B.6 contain the results of the first phase of the methodology. More specifically, there is a presentation of the results of the four MCDA methods for each sector. The first phase is completed with the cumulative ranking of the securities which is presented in table B.7. The results of the second phase (portfolio optimisation) are included in four tables. Firstly, the pareto optimal portfolios of the MIQP mean - variance method are presented in table B.8. Secondly, the goal programming portfolio is presented in table B.9. Table B.10 contains the pareto optimal portfolios for the MOIP PROMETHEE flow method. Finally, the portfolio produced by the genetic algorithm is presented in table B.11.

2. **Paris stock exchange**: In tables B.12 - B.14 the input matrix for the three industrial segments is presented. Tables B.15 - B.17 contain the results of the first phase of the methodology. More specifically, there is a presentation of the results of the four MCDA methods for each sector. The first phase is completed with the cumulative ranking of the securities which is presented in table B.18. The results of the second phase (portfolio optimisation) are included in four tables. Firstly, the pareto optimal portfolios of the MIQP mean - variance method are presented in table B.19. Secondly, the goal programming portfolio is presented in table B.20. Table B.21 contains the pareto optimal portfolios for the MOIP PROMETHEE flow method. Finally, the portfolio produced by the genetic algorithm is presented in table B.22.

3. **Tokyo stock exchange**: In tables B.23 - B.25 the input matrix for the three industrial segments is presented. Tables B.26 - B.28 contain the results of the

first phase of the methodology. More specifically, there is a presentation of the results of the four MCDA methods for each sector. The first phase is completed with the cumulative ranking of the securities which is presented in table B.29. The results of the second phase (portfolio optimisation) are included in four tables. Firstly, the pareto optimal portfolios of the MIQP mean - variance method are presented in table B.30. Secondly, the goal programming portfolio is presented in table B.31. Table B.32 contains the pareto optimal portfolios for the MOIP PROMETHEE flow method. Finally, the portfolio produced by the genetic algorithm is presented in table B.33.

NASDAQ stock exchange

Phase I: Portfolio Selection

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Bel Fuse A	9.41	1.38	0.542	2.09	1.85	1	-5.45	-15.44
Cognizant A	16.52	3.6	16.46	1.05	1.35	1	-6.36	-35.71
Activision Blizzard	29.38	1.82	7.11	0.86	0.69	3	14.73	-27.65
Formula Systems ADR	25.69	2.4	1.56	0.87	1.4	5	72.7	47.14
LM Ericsson B ADR	1320	0.01	23.17	0.54	1.31	1	-6.99	0
Allied Motion	19.41	1.74	0.34	1.52	0.35	2	-24.28	-24.36
Amdocs	22.62	2.9	4.06	0.39	1.74	5	12.12	7.58
American Software	74.12	0.21	0.108	0.5	2.84	5	48.33	34.43
Analog Devices	26.55	4.07	7.69	1.44	2	5	25.91	29.16
Apple	19.72	11.51	259.03	1.22	1.36	5	43.93	5.87
Applied Materials	16.96	3	14.87	1.63	1.65	5	55.28	55.05
Jack Henry&Associates	40.5	3.51	1.55	0.9	1.12	5	12.5	-3.38
AstroNova	17.16	0.89	0.141	0.4	1.83	1	-18.35	-24.95
...	...	...	...	...	...	...	...	...
Xperi	39.28	0.51	0.408	0.36	4.03	1	7.94	46.71

Table B.1: Evaluation Table input data for NASDAQ Technology Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Diamondback	13.44	6.25	3.05	0.81	0.89	1	-9.32	-33.86
Alliance Resource	3.8	3.56	2.07	0.97	15.95	1	-22.09	-31.77
Viper Energy Ut	38.1	0.68	0.284	1.33	7.25	1	-0.46	-30.38
Dorchester Minerals	11.23	1.59	0.075	0.98	10.69	1	21.79	-6.6
Hallador	11.15	0.3	0.328	0.04	4.85	1	-34.71	-45.56
TransGlobe Energy	4.84	0.26	0.164	1.18	3.39	1	-32.62	-57.86

Table B.2: Evaluation Table input data for NASDAQ Energy Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Nuveen NAS. 100 Dyn. Ov	23.9	0.94	0.01	1.15	6.93	4	12.35	-2.01
1st Source	13.93	3.32	0.19	1.05	2.34	3	14.45	-5.47
1st Constitution Bancorp	11.79	1.62	0.032	0.25	1.57	5	-4.26	-2
Bancorp 34	37.88	0.39	0.009	0.58	1.36	2	-0.47	-6.24
National General A Pref	11.29	2.01	4.6	0.62	0.88	5	25.99	8.84
Donegal A	29.34	0.49	0.798	0.25	4.03	3	5.57	2.86
ACNB	10.13	3.36	0.043	0.17	2.94	2	-13.38	-11.11
Hennessy Ad	8.4	1.33	0.045	0.9	4.94	2	11.29	-17.48
Grupo Financiero Galicia ADR	3.51	3.66	2.39	1.31	2.82	1	-53.39	-42.76
Alcentra Capital Corp	14.11	0.63	0.026	0.59	8.08	5	38.33	50.42
Alerus Fin	12.93	1.65	0.101	0.6	2.62	3	11.32	-4.76
Amark Preci	36.14	0.31	4.78	0.01	2.83	1	-4.07	-16.41
America First Tax	12.92	0.6	0.088	0.36	6.45	5	37.9	40.14
...	...	...	...	...	...	...	...	...
Zions	10.75	4.15	1.62	1.56	3.05	3	9.62	-5.68

Table B.3: Evaluation Table input data for NASDAQ Financial Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Diamondback	0.77	50.50	11.92	57.86
Alliance Resource	3.88	52.91	26.01	56.85
Viper Energy Ut	-3.17	21.31	-22.68	31.89
Dorchester Minerals	2.29	49.09	19.89	54.99
Hallador	-0.67	29.75	-14.36	34.57
TransGlobe Energy	-3.10	28.99	-20.79	30.76

Table B.4: Phase I results for NASDAQ Energy Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Bel Fuse A	-68.79	21.87	-37.96	40.79
Cognizant A	-48.52	28.27	-23.47	42.11
Activision Blizzard	-21.95	35.01	-17.25	41.47
Formula Systems ADR	36.06	53.34	25.57	46.41
LM Ericsson B ADR	-66.26	20.62	-29.16	20.38
Allied Motion	-54.09	24.66	-37.84	39.96
Amdocs	27.48	48.24	10.71	43.90
American Software	34.78	50.44	20.11	44.68
Analog Devices	10.52	47.38	9.78	45.11
Apple	72.58	67.99	31.36	67.67
Applied Materials	16.15	50.06	17.28	46.87
Jack Henry&Associates	5.26	44.65	0.74	42.77
...	...	...	...	...
Xperi	-4.21	37.64	7.71	44.82

Table B.5: Phase I results for NASDAQ Technology Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Nuveen NASDAQ 100 Dynamic Ov	-0.52	42.37	2.04	41.06
1st Source	-73.66	37.21	-7.02	42.17
1st Constitution Bancorp	33.58	44.49	2.28	41.25
Bancorp 34	-102.57	31.59	-21.52	38.32
National General A Pref	240.82	49.91	24.42	47.76
Donegal A	48.64	41.03	14.12	41.09
ACNB	-23.80	35.54	-9.09	41.03
Hennessy Ad	-69.23	34.90	-9.49	41.15
Grupo Financiero Galicia ADR	52.31	22.39	-28.24	39.43
Alcentra Capital Corp	211.38	57.17	37.45	47.38
Alerus Fin	-41.63	38.70	-3.80	41.81
Amark Preci	156.21	34.77	-3.52	41.67
...	...	...	...	...
Zions	102.93	36.54	2.13	43.32

Table B.6: Phase I results for NASDAQ Financial Sector

Technology	**Energy**	**Financial**
Apple	Alliance Resource	Cincinnati Financial
Microsoft	Diamondback	Willis Towers Watson
AudioCodes	Dorchester Minerals	Erie Indemnity
KLA-Tencor	Hallador	Nasdaq Inc
TTEC	TransGlobe Energy	T Rowe
Simulations Plus	Viper Energy Ut	Alcentra Capital Corp
Equinix		Ares Capital
Wayside		CME Group
CDW Corp		Principal Financial
Teradyne		MarketAxesss
Sapiens		America First Tax
Formula Systems ADR		LPL Financial
NXP		First Capital
Intel		The Carlyle
Seagate		Verisk
Texas Instruments		National General A Pref
CSG Systems		New York Mortgage Pref
Garmin		New York Mortgage Trust Inc Pref
NIC		Selective
Elbit Systems		Safety Insurance

Table B.7: Selected securities from NASDAQ stock exchange

Phase II: Portfolio Optimisation

Method 1: Mean - Variance MIQP model

Portf	AAPL	MSFT	AUDC	KLAC	TTEC	SLP	EQIX	WSTG	CDW	TER	SPNS	...	NYMT	NYMTP	SIGI	SAFT
1	0.0	0.00	0.00	0.0	0.0	0.01	0.04	0.07	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
2	0.0	0.00	0.00	0.0	0.0	0.01	0.04	0.06	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
3	0.0	0.00	0.00	0.0	0.0	0.01	0.04	0.06	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
4	0.0	0.00	0.00	0.0	0.0	0.01	0.04	0.06	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
5	0.0	0.00	0.01	0.0	0.0	0.01	0.04	0.05	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
6	0.0	0.00	0.01	0.0	0.0	0.01	0.04	0.05	0.00	0.0	0.0	...	0.0	0.16	0.00	0.07
7	0.0	0.00	0.01	0.0	0.0	0.02	0.03	0.04	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
8	0.0	0.00	0.02	0.0	0.0	0.02	0.03	0.04	0.00	0.0	0.0	...	0.0	0.16	0.00	0.06
9	0.0	0.00	0.02	0.0	0.0	0.02	0.03	0.04	0.00	0.0	0.0	...	0.0	0.16	0.00	0.07
10	0.0	0.00	0.02	0.0	0.0	0.02	0.03	0.03	0.00	0.0	0.0	...	0.0	0.16	0.00	0.07
11	0.0	0.00	0.03	0.0	0.0	0.02	0.03	0.03	0.00	0.0	0.0	...	0.0	0.16	0.00	0.07
12	0.0	0.00	0.03	0.0	0.0	0.02	0.03	0.03	0.00	0.0	0.0	...	0.0	0.15	0.01	0.07
13	0.0	0.00	0.03	0.0	0.0	0.03	0.02	0.02	0.00	0.0	0.0	...	0.0	0.15	0.01	0.07
14	0.0	0.00	0.04	0.0	0.0	0.03	0.02	0.02	0.00	0.0	0.0	...	0.0	0.15	0.01	0.07
15	0.0	0.00	0.04	0.0	0.0	0.03	0.02	0.02	0.00	0.0	0.0	...	0.0	0.15	0.01	0.07
16	0.0	0.00	0.04	0.0	0.0	0.03	0.02	0.01	0.00	0.0	0.0	...	0.0	0.15	0.02	0.06
17	0.0	0.00	0.05	0.0	0.0	0.03	0.02	0.01	0.00	0.0	0.0	...	0.0	0.15	0.01	0.06
18	0.0	0.00	0.05	0.0	0.0	0.03	0.01	0.01	0.00	0.0	0.0	...	0.0	0.15	0.02	0.06
19	0.0	0.00	0.05	0.0	0.0	0.03	0.01	0.00	0.00	0.0	0.0	...	0.0	0.14	0.02	0.06
20	0.0	0.00	0.05	0.0	0.0	0.03	0.01	0.00	0.01	0.0	0.0	...	0.0	0.14	0.02	0.06
21	0.0	0.00	0.06	0.0	0.0	0.04	0.00	0.00	0.00	0.0	0.0	...	0.0	0.14	0.03	0.06
22	0.0	0.00	0.06	0.0	0.0	0.04	0.00	0.00	0.00	0.0	0.0	...	0.0	0.14	0.03	0.06
23	0.0	0.00	0.07	0.0	0.0	0.04	0.00	0.00	0.00	0.0	0.0	...	0.0	0.13	0.03	0.05
24	0.0	0.00	0.07	0.0	0.0	0.04	0.00	0.00	0.01	0.0	0.0	...	0.0	0.12	0.04	0.05
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
47	0.0	0.00	0.41	0.0	0.0	0.13	0.00	0.00	0.00	0.0	0.0	...	0.0	0.00	0.00	0.00
48	0.0	0.00	0.45	0.0	0.0	0.14	0.00	0.00	0.00	0.0	0.0	...	0.0	0.00	0.00	0.00
49	0.0	0.00	0.49	0.0	0.0	0.14	0.00	0.00	0.00	0.0	0.0	...	0.0	0.00	0.00	0.00
50	0.0	0.00	0.50	0.0	0.0	0.20	0.00	0.00	0.00	0.0	0.0	...	0.0	0.00	0.00	0.00

Table B.8: Pareto optimal portfolios for NASDAQ securities

Method 2: Goal programming model

AAPL	MSFT	AUDC	KLAC	TTEC	SLP	EQIX	WSTG	CDW	TER
0.2	0.1074	0.03	0.0	0.03	0.0411	0.03	0.0	0.0	0.0
SPNS	FORTY	NXPI	INTC	STX	TXN	CSGS	GRMN	EGOV	ESLT
0.03	0.0	0.0	0.1714	0.0	0.0	0.0	0.0	0.0	0.03
ARLP	FANG	DMLP	HNRG	TGA	VNOM	CINF	WLTW	ERIE	NDAQ
0.0	0.03	0.0	0.0	0.0	0.0	0.03	0.03	0.03	0.03
TROW	ABDC	ARCC	CME	PFG	MKTX	ATAX	LPLA	FCAP	CG
0.0	0.0	0.0	0.03	0.0	0.03	0.0	0.0	0.03	0.0
VRSK	NGHC	NYMT	NYMTP	SIGI	SAFT				
0.03	0.03	0.0	0.0	0.03	0.0				

Table B.9: Goal programming portfolio for NASDAQ stock exchange

Method 3: PROMETHEE flow multiobjective programming model

Portf	AAPL	MSFT	AUDC	KLAC	TTEC	SLP	EQIX	WSTG	CDW	TER	SPNS	...	NYMT	NYMTP	SIGI	SAFT
1	0.00	0.00	0.03	0.00	0.03	0.20	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
2	0.00	0.00	0.03	0.00	0.03	0.20	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
3	0.00	0.00	0.03	0.00	0.03	0.17	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
4	0.00	0.00	0.03	0.00	0.03	0.13	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
5	0.00	0.00	0.03	0.00	0.03	0.08	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
6	0.00	0.00	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
7	0.00	0.00	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.03
8	0.00	0.00	0.03	0.00	0.03	0.03	0.03	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.05
9	0.00	0.00	0.03	0.00	0.03	0.03	0.00	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.07
10	0.03	0.00	0.03	0.00	0.00	0.03	0.00	0.03	0.00	0.00	0.00	...	0.00	0.00	0.03	0.08
11	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.03	0.00	0.00	0.00	...	0.03	0.00	0.03	0.09
12	0.03	0.00	0.03	0.03	0.00	0.00	0.00	0.03	0.00	0.00	0.00	...	0.00	0.00	0.03	0.08
13	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.09
14	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.05
15	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
16	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
17	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
18	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
19	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.00	0.00	0.03	0.03
20	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	...	0.03	0.03	0.03	0.03
21	0.03	0.03	0.00	0.04	0.00	0.00	0.00	0.00	0.03	0.00	0.00	...	0.00	0.00	0.03	0.03
22	0.03	0.03	0.00	0.08	0.00	0.00	0.00	0.00	0.03	0.00	0.00	...	0.00	0.00	0.03	0.03
23	0.03	0.03	0.00	0.09	0.00	0.00	0.00	0.00	0.00	0.03	0.00	...	0.00	0.00	0.00	0.03
24	0.03	0.03	0.00	0.10	0.00	0.00	0.00	0.00	0.03	0.03	0.00	...	0.00	0.00	0.00	0.03
25	0.03	0.03	0.00	0.14	0.00	0.00	0.00	0.00	0.03	0.03	0.00	...	0.00	0.00	0.00	0.03
26	0.03	0.03	0.00	0.18	0.00	0.00	0.00	0.00	0.03	0.03	0.00	...	0.00	0.00	0.00	0.03
27	0.03	0.03	0.00	0.20	0.00	0.00	0.00	0.00	0.03	0.03	0.00	...	0.00	0.00	0.00	0.03
28	0.03	0.03	0.00	0.20	0.00	0.00	0.00	0.00	0.03	0.03	0.00	...	0.00	0.00	0.03	0.03
29	0.03	0.03	0.00	0.20	0.00	0.00	0.00	0.00	0.03	0.03	0.03	...	0.00	0.00	0.00	0.03
30	0.03	0.03	0.00	0.20	0.00	0.00	0.00	0.00	0.03	0.03	0.03	...	0.00	0.00	0.00	0.00

Table B.10: Set of efficient portfolios for NASDAQ stock exchange with MOIP PROMETHEE method

Method 4: Genetic algorithm model

AAPL	MSFT	AUDC	KLAC	TTEC	SLP	EQIX	WSTG	CDW	TER
0.0173	0.0262	0.0138	0.0549	0.0331	0.0193	0.0178	0.0333	0.031	0.0123
SPNS	FORTY	NXPI	INTC	STX	TXN	CSGS	GRMN	EGOV	ESLT
0.0131	0.0131	0.0104	0.0419	0.0137	0.0118	0.0101	0.0314	0.0164	0.0164
ARLP	FANG	DMLP	HNRG	TGA	VNOM	CINF	WLTW	ERIE	NDAQ
0.0252	0.0225	0.0404	0.028	0.03	0.0148	0.0331	0.0134	0.0101	0.0166
TROW	ABDC	ARCC	CME	PFG	MKTX	ATAX	LPLA	FCAP	CG
0.0214	0.0217	0.0194	0.0232	0.0283	0.011	0.0126	0.0283	0.0163	0.0117
VRSK	NGHC	NYMT	NYMTP	SIGI	SAFT				
0.0109	0.019	0.0255	0.0135	0.0394	0.0264				

Table B.11: Genetic algorithm portfolio for NASDAQ stock exchange

Paris stock exchange

Phase I: Portfolio Selection

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Total	12.91	3.56	165.26	0.93	5.63	1	0.74	-11.94
TechnipFMC	15.76	4.26	11.44	1.02	2.14	1	17.85	-19.72
Rubis	17.97	2.86	5.08	0.59	3.1	5	9.32	13.94
GTT	25.92	3.32	0.241	0.71	3.82	5	28.15	32.38
Total Gabon	2.77	47.45	0.78	0.74	7.47	1	5.6	-8.97
Maurel et Prom	8.75	0.31	0.597	0.83	1.45	1	-15.33	-36.54
Docks des Petroles dAmbes	17.88	26.41	0.016	0.4	6.35	2	4.42	-5.22

Table B.12: Evaluation Table input data for Paris Energy Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Akka	20.21	2.87	0.793	1.24	1.2	4	31.45	5.25
Alten	22.32	4.67	2.46	0.92	0.96	5	43.47	27.98
Artois Nom.	109.29	44.1	0.156	0.41	0.71	1	-4.55	-18.99
Atos	12.24	5.46	12	0.79	2.54	3	25.91	-7.41
Aubay	16.57	2.07	0.203	0.58	1.92	5	21.85	3.94
Aures Tech	14.92	1.15	0.05	0.43	5.82	1	-41.76	-52.08
Axway	238.64	0.04	0.287	0.91	3.83	1	-15.86	-33.86
Capgemini	23.47	4.69	13.74	1.06	1.54	4	26.84	6.48
Cofidur	3.44	80.91	0.077	0.12	2.88	1	-18.24	-21.02
Coheris	28.88	0.08	0.013	0.28	1.34	3	36.59	24.44
CS Communication	26.16	0.16	0.216	0.46	0.96	1	0.97	-15.31
...	...	...	...	...	...	...	...	...
Wedia	19.42	1.19	0.005	0.53	1.3	3	0.87	-10.81

Table B.13: Evaluation Table input data for Paris Technology Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
BNP Paribas	7.75	6.06	73.76	1.37	6.43	2	19.33	-3.32
AXA	38.48	0.62	125.06	1.3	5.29	5	27.04	7.11
Credit Agricole	8.54	1.35	68.7	1.48	5.46	3	22.53	-2.83
Societe Generale	7.01	3.71	43.29	1.46	8.45	1	-6.11	-24.18
Amundi	14.61	4.37	5.78	1.62	3.92	5	38.21	13.6
Natixis	7.39	0.54	15.46	1.46	9.34	1	-3.79	-23.73
CNP Assurances	9.33	1.92	16.44	0.82	4.69	1	-3.51	-7.89
SCOR	19.73	1.88	16.17	0.72	4.44	3	-6.32	-10.43
Euronext	23.63	3.03	0.775	0.84	2.42	5	42.15	29.06
Eurazeo	21.31	3.21	4.5	0.74	1.74	4	6.52	3.76
FFP	20.02	5.29	0.183	1.35	1.89	5	30.92	1.74
...	...	...	...	...	...	...	...	...
Idsud	96.17	0.74	0.001	0.46	0.28	5	33.64	9.16

Table B.14: Evaluation Table input data for Paris Financial Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Total	1.87	39.96	1.76	46.01
TechnipFMC	-16.22	21.42	-24.20	35.46
Rubis	4.92	46.19	7.28	45.95
GTT	-6.69	49.49	14.10	55.19
Total Gabon	2.83	54.22	20.84	47.81
Maurel et Prom	14.57	13.15	-40.29	15.43
Docks des Petroles dAmbes	-1.28	48.42	20.49	41.14

Table B.15: Phase I results for Paris Energy Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Akka	-8.95	39.54	2.10	41.81
Alten	12.35	49.95	22.23	45.65
Artois Nom.	-13.62	30.72	-15.37	38.05
Atos	31.67	44.56	13.85	45.87
Aubay	9.84	46.75	13.76	42.24
Aures Tech	-23.84	24.50	-25.69	35.43
Axway	-45.28	12.52	-42.77	13.53
Capgemini	31.90	47.02	16.59	46.00
Cofidur	24.54	44.05	-1.61	53.25
Coheris	6.90	45.37	18.61	43.53
CS Communication	-18.40	29.17	-15.32	37.36
...	...	...	...	...
Wedia	-9.33	36.14	-4.87	38.62

Table B.16: Phase I results for Paris Technology Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
BNP Paribas	-5.60	45.84	3.74	48.68
AXA	6.35	54.93	3.69	52.32
Credit Agricole	-7.09	44.78	0.47	46.53
Societe Generale	-6.66	33.65	-18.29	38.58
Amundi	-10.83	47.59	3.72	44.36
Natixis	-1.62	30.83	-25.20	36.19
CNP Assurances	-16.06	30.40	-22.09	37.35
SCOR	-14.24	34.55	-24.01	34.88
Euronext	-1.59	49.61	3.92	47.34
Eurazeo	-12.72	37.39	-23.18	37.23
FFP	-16.52	42.72	-11.08	40.53
...	...	...	...	...
Idsud	-19.58	33.78	-15.05	34.12

Table B.17: Phase I results for Paris Financial Sector

Technology	**Energy**	**Financial**
Schneider Electric	Total Gabon	Crcam Sud RA
Ingenico	GTT	CRCAM Atlantique
Legrand	Rubis	Crcam Touraine
Evolis	Docks des Petroles dAmbes	Crcam Norm.Sei
Tessi	Total	Crcam Morbihan
GEA	Maurel et Prom	Ca Toulouse 31 CCI
Capgemini	TechnipFMC	Coface
Harvest		Crcam Loire Ht
Environnement		AXA
Alten		CRCAM Brie Picardie 2
Somfy		Altamir
Atos		Crcam Ille-Vil
Dassault Systemes		CRCAM Langued
Cofidur		Euronext
Perrier Gerard		BNP Paribas
Coheris		Groupe IRD
Thales		ABC Arbitrage
Esker		April
SPII		Amundi
Aubay		Credit Agricole

Table B.18: Selected securities from Paris stock exchange

Phase II: Portfolio Optimisation

Method 1: Mean - Variance MIQP model

Portf	SU	ING	LR	ALTVO	GEA	CAP	ALTEV	ATE	SO	ATO	DSY	...	IRD	ABCA	AMUN	ACA
1	0.0	0.0	0.02	0.04	0.07	0.0	0.03	0.0	0.0	0.0	0.00	...	0.12	0.03	0.0	0.0
2	0.0	0.0	0.02	0.04	0.07	0.0	0.03	0.0	0.0	0.0	0.00	...	0.12	0.03	0.0	0.0
3	0.0	0.0	0.01	0.03	0.07	0.0	0.03	0.0	0.0	0.0	0.00	...	0.13	0.03	0.0	0.0
4	0.0	0.0	0.01	0.03	0.07	0.0	0.03	0.0	0.0	0.0	0.01	...	0.13	0.03	0.0	0.0
5	0.0	0.0	0.00	0.03	0.07	0.0	0.03	0.0	0.0	0.0	0.01	...	0.13	0.03	0.0	0.0
6	0.0	0.0	0.00	0.02	0.06	0.0	0.03	0.0	0.0	0.0	0.01	...	0.13	0.03	0.0	0.0
7	0.0	0.0	0.00	0.02	0.06	0.0	0.03	0.0	0.0	0.0	0.01	...	0.13	0.03	0.0	0.0
8	0.0	0.0	0.00	0.02	0.06	0.0	0.02	0.0	0.0	0.0	0.01	...	0.14	0.03	0.0	0.0
9	0.0	0.0	0.00	0.01	0.06	0.0	0.02	0.0	0.0	0.0	0.00	...	0.14	0.02	0.0	0.0
10	0.0	0.0	0.00	0.01	0.06	0.0	0.02	0.0	0.0	0.0	0.00	...	0.15	0.02	0.0	0.0
11	0.0	0.0	0.00	0.00	0.05	0.0	0.02	0.0	0.0	0.0	0.00	...	0.15	0.02	0.0	0.0
12	0.0	0.0	0.00	0.00	0.05	0.0	0.02	0.0	0.0	0.0	0.00	...	0.15	0.02	0.0	0.0
13	0.0	0.0	0.00	0.00	0.05	0.0	0.02	0.0	0.0	0.0	0.00	...	0.16	0.02	0.0	0.0
14	0.0	0.0	0.00	0.00	0.04	0.0	0.01	0.0	0.0	0.0	0.01	...	0.17	0.01	0.0	0.0
15	0.0	0.0	0.00	0.00	0.04	0.0	0.01	0.0	0.0	0.0	0.01	...	0.17	0.01	0.0	0.0
16	0.0	0.0	0.00	0.00	0.03	0.0	0.01	0.0	0.0	0.0	0.01	...	0.18	0.01	0.0	0.0
17	0.0	0.0	0.00	0.00	0.03	0.0	0.01	0.0	0.0	0.0	0.01	...	0.19	0.01	0.0	0.0
18	0.0	0.0	0.00	0.00	0.03	0.0	0.00	0.0	0.0	0.0	0.01	...	0.20	0.01	0.0	0.0
19	0.0	0.0	0.00	0.00	0.02	0.0	0.00	0.0	0.0	0.0	0.01	...	0.20	0.00	0.0	0.0
20	0.0	0.0	0.00	0.00	0.02	0.0	0.00	0.0	0.0	0.0	0.01	...	0.21	0.00	0.0	0.0
21	0.0	0.0	0.00	0.00	0.01	0.0	0.00	0.0	0.0	0.0	0.01	...	0.22	0.00	0.0	0.0
22	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.01	...	0.23	0.00	0.0	0.0
23	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.01	...	0.24	0.00	0.0	0.0
24	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.01	...	0.25	0.00	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
47	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.00	...	0.50	0.00	0.0	0.0
48	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.00	...	0.50	0.00	0.0	0.0
49	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.00	...	0.50	0.00	0.0	0.0
50	0.0	0.0	0.00	0.00	0.00	0.0	0.00	0.0	0.0	0.0	0.00	...	0.50	0.00	0.0	0.0

Table B.19: Pareto optimal portfolios for Paris securities

Method 2: Goal programming model

SU	ING	LR	ALTVO	GEA	CAP	ALTEV	ATE	SO	ATO
0.03	0.0392	0.03	0.0	0.03	0.03	0.03	0.2	0.03	0.03
DSY	ALCOF	PERR	COH	HO	SII	EC	GTT	RUI	FTI
0.2	0.0	0.0	0.03	0.03	0.0807	0.0	0.0	0.0	0.03
CRSU	CRAV	CRTO	CCN	CMO	CAT31	COFA	CRLO	CS	LTA
0.0	0.0	0.03	0.0	0.03	0.0	0.0	0.0	0.03	0.0
CIV	CRLA	ENX	BNP	IRD	ABCA	AMUN	ACA		
0.0	0.0	0.03	0.03	0.0	0.0	0.0	0.03		

Table B.20: Goal programming portfolio for Paris stock exchange

Method 3: PROMETHEE flow multiobjective programming model

Portf	SU	ING	LR	ALTVO	GEA	CAP	ALTEV	ATE	SO	ATO	DSY	...	IRD	ABCA	AMUN	ACA
1	0.00	0.00	0.00	0.03	0.03	0.00	0.03	0.00	0.00	0.00	0.00	...	0.20	0.03	0.00	0.00
2	0.00	0.00	0.03	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.00	...	0.03	0.03	0.00	0.00
3	0.00	0.00	0.03	0.03	0.09	0.00	0.03	0.00	0.03	0.00	0.00	...	0.03	0.03	0.00	0.00
4	0.00	0.03	0.03	0.03	0.11	0.00	0.03	0.00	0.03	0.00	0.03	...	0.00	0.03	0.00	0.00
5	0.03	0.03	0.03	0.03	0.09	0.00	0.03	0.00	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
6	0.03	0.03	0.03	0.08	0.20	0.00	0.03	0.00	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
7	0.03	0.03	0.03	0.12	0.17	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
8	0.03	0.03	0.04	0.20	0.08	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
9	0.04	0.03	0.08	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
10	0.11	0.03	0.03	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
11	0.14	0.03	0.03	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
12	0.17	0.03	0.03	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
13	0.20	0.03	0.03	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
14	0.20	0.03	0.08	0.20	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
15	0.20	0.03	0.14	0.15	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
16	0.20	0.04	0.20	0.08	0.03	0.00	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
17	0.20	0.06	0.20	0.06	0.03	0.03	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.00	0.00
18	0.20	0.10	0.19	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.00	0.00
19	0.20	0.09	0.20	0.03	0.03	0.03	0.03	0.03	0.03	0.00	0.03	...	0.00	0.00	0.03	0.00
20	0.20	0.19	0.10	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.03	0.00
21	0.20	0.16	0.13	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.03	0.00
22	0.20	0.18	0.11	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.03	0.00
23	0.20	0.20	0.09	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.03	0.03
24	0.20	0.20	0.09	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.03	0.03
25	0.20	0.20	0.05	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.07	0.03
26	0.20	0.18	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.11	0.03
27	0.20	0.12	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.17	0.03
28	0.20	0.05	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.20	0.07
29	0.16	0.03	0.03	0.03	0.00	0.03	0.03	0.03	0.03	0.03	0.03	...	0.00	0.00	0.20	0.13
30	0.03	0.03	0.03	0.00	0.00	0.03	0.00	0.03	0.03	0.03	0.03	...	0.00	0.00	0.20	0.20

Table B.21: Set of efficient portfolios for Paris stock exchange with MOIP PROMETHEE method

Method 4: Genetic algorithm model

SU	ING	LR	ALTVO	GEA	CAP	ALTEV	ATE	SO	ATO
0.0004	0.0005	0.1471	0.244	0.0008	0.0025	0.0042	0.0991	0.0026	0.0019
DSY	ALCOF	PERR	COH	HO	SII	EC	GTT	RUI	FTI
0.0032	0.0025	0.0002	0.0	0.002	0.003	0.0028	0.0017	0.0009	0.0003
CRSU	CRAV	CRTO	CCN	CMO	CAT31	COFA	CRLO	CS	LTA
0.2517	0.0028	0.0015	0.0002	0.0035	0.0025	0.0037	0.1904	0.0038	0.0033
CIV	CRLA	ENX	BNP	IRD	ABCA	AMUN	ACA		
0.0031	0.0028	0.0033	0.0005	0.0032	0.0002	0.0025	0.0011		

Table B.22: Genetic algorithm portfolio for Paris stock exchange

Tokyo stock exchange

Phase I: Portfolio Selection

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
San-Ai Oil	9.49	112.45	749.1	1.11	2.53	3	-3.4	-17.62
BP Castrol KK	20.06	72.08	12.85	1.01	5.39	3	17.58	-6.14
Mitsui Matsushima Co Ltd	7.67	160.3	73.95	0.6	4.07	1	-9.69	-36.17
Idemitsu Kosan Co Ltd	12.78	242.52	4896.74	1.01	3.23	1	-15.37	-43.43
Sinanen Co Ltd	8.82	211.02	242.78	0.65	4.03	3	-23	-29.19
Itochu Enex Co Ltd	8.16	103.79	1001	0.79	4.96	1	-12.58	-22.99
Toell Co Ltd	15.17	52.01	23.67	0.98	1.9	1	20.03	-11.92
Nippon Coke & Eng Ltd	8.02	10.47	119.91	1.02	3.57	1	-8.79	-24.55
Inpex Corp.	12.77	74.64	1056.44	1.43	1.89	1	-3.22	-32.35
...	...	...	...	...	...	...	...	...
Sanrin	11.06	65.47	28.89	-0.17	2.62	4	-1.37	0.14

Table B.23: Evaluation Table input data for Tokyo Energy Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
Yaskawa Electric Corp.	42	90.71	438.17	1.36	1.36	5	41.22	19.44
Advantest Corp.	18.01	278.17	277.69	1.33	1.84	5	123.26	130.56
Rohm Ltd	24.25	350.57	388.65	1.3	1.76	5	20.74	14.56
Hitachi High-Technologies Corp	18.42	346.36	719.69	1.01	1.65	5	84.66	76.49
Nitto Denko Co	14.22	380.76	983.73	1.55	3.32	1	-2.31	-31.86
Shimadzu Corp	24.68	107.21	386.33	1.13	1.06	3	21.71	-17.95
Otsuka Corp	20.21	202.17	820.16	1.07	2.08	5	35.26	5.42
Disco Corp	29.33	726.98	139.91	1.49	1.51	5	65.91	21.83
...	...	...	...	...	...	...	...	...
Obic Co Ltd	32.02	386.97	75.92	0.75	1.09	5	45.94	16.89

Table B.24: Evaluation Table input data for Tokyo Technology Sector

Decision Matrix	P/E Ratio	EPS	Rev (B)	Beta	DY(%)	Mon	YTD (%)	1 Year
The 77 Bank Ltd	7.98	206.74	58.19	1.3	2.88	1	-15.31	-33.29
Nihon M&A Center	51.39	58.05	30.2	0.64	0.77	5	31.81	-5.93
Acom Co Ltd	15.05	29.03	278.75	0.91	0.46	5	19.5	0.47
Activia Properties	28.93	19700	28.66	0.08	3.44	5	30.26	20.62
MS&AD Insurance Group Hold.	9.77	351.17	5318.69	1.28	4.08	3	9.51	-4.24
Advance Create	21.46	82	10.22	0.95	2.84	3	4.79	-19.07
Japan Investment Adviser	12.7	153.63	14.9	0.97	0.97	1	-37.34	-51.8
Aeon Financial Service Co Ltd	9.56	172.3	241.27	1.43	4.13	1	-17.2	-29
Aichi Bank Ltd	8.8	409.5	26.6	1.11	2.77	1	-5.15	-21.93
...	...	...	...	...	...	...	...	...
United Urban	27.14	7920	52.99	0.15	3.32	5	27.44	24.3

Table B.25: Evaluation Table input data for Tokyo Financial Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
San-Ai Oil	-8.26	35.41	-1.98	36.82
BP Castrol KK	-7.03	43.83	-0.41	40.18
Mitsui Matsushima Co Ltd	-7.30	34.96	-8.47	34.79
Idemitsu Kosan Co Ltd	6.25	34.21	-7.62	36.75
Sinanen Co Ltd	-5.42	40.88	3.32	35.57
Itochu Enex Co Ltd	1.08	36.50	0.41	36.55
Toell Co Ltd	-14.18	28.83	-24.27	38.09
Nippon Coke & Engineering Ltd	-9.85	27.98	-25.99	33.67
Inpex Corp.	-7.87	21.78	-29.78	31.25
...	...	...	...	...
Sanrin	8.87	48.76	9.52	42.24

Table B.26: Phase I results for Tokyo Energy Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
Yaskawa Electric Corp.	81.74	40.58	15.29	40.78
Advantest Corp.	362.00	49.74	36.81	45.76
Rohm Ltd	244.29	42.03	23.33	41.37
Hitachi High-Technologies Corp	377.45	48.11	49.79	44.51
Nitto Denko Co	106.09	30.57	-2.83	41.25
Shimadzu Corp	-67.24	32.75	-5.57	39.99
Otsuka Corp	162.07	43.95	31.30	41.54
Disco Corp	327.65	43.86	26.62	42.84
Trend Micro Inc.	-70.51	36.76	-4.64	39.50
...	...	...	...	...
Obic Co Ltd	323.88	43.46	30.73	41.28

Table B.27: Phase I results for Tokyo Technology Sector

Name	ELECTRE 3	MAUT	PROMETHEE	TOPSIS
The 77 Bank Ltd	-48.24	22.83	-17.64	37.85
Nihon M&A Center	44.83	38.46	12.44	38.75
Acom Co Ltd	34.99	37.18	10.81	40.46
Activia Properties	117.69	57.13	52.13	46.74
MS&AD Insurance Group Holdings	107.35	38.76	32.79	43.58
Advance Create	-21.52	32.48	0.86	38.62
Japan Investment Adviser	-68.94	20.18	-33.76	36.34
Aeon Financial Service Co Ltd	-43.10	24.18	-12.88	38.07
Aichi Bank Ltd	41.38	24.99	-3.73	38.54
...	...	...	...	...
United Urban	111.27	51.50	50.83	43.39

Table B.28: Phase I results for Tokyo Financial Sector

Technology	**Energy**	**Financial**
Tokyo Electron	Mitsui	Kenedix Office
NEC Corp.	JP Petroleum Exploration Ltd	Activia Properties
Mitsubishi Electric	Mitsuuroko Group Holdings	Daiwa Office
Ryoyu Systems	Daimaru Enawin	Tosei REIT
Hitachi	Toa Oil	Kenedix Retail Reit
Hitachi High-Technologies Corp	Iwatani Corp	LaSalle Logiport
Advantest Corp.	Marubeni Corp.	Takara Leben Infrastructure
Oricon	Sanrin	United Urban
NuFlare Tech	Sumiseki Holdings Inc	Wealth Management
PCA Corp	Tokai Holdings Corp	Sekisui House Reit
SMC Corp	Cosmo Energy Holdings	Tokio Marine Holdings, Inc.
Synclayer	Shinko Plantech	Ichigo Hotel REIT
Holon	JX Holdings, Inc.	Healthcare Medical Invest
Softmax	Sala Corp	Hankyu REIT
Mamezou Holdings	BP Castrol KK	Nippon Healthcare
Lasertec Corp	Sojitz Corp.	MS&AD Insurance Group Holdings
Nissin Electric	Kamei Corp	NEC Capital Solutions
Applied Tech	Itochu Enex Co Ltd	Newton Financial Consulting
Uchida Yoko Co Ltd	Sinanen Co Ltd	JACCS Co Ltd
TDK	Idemitsu Kosan Co Ltd	NKSJ Holdings, Inc.

Table B.29: Selected securities from Tokyo stock exchange

Phase II: Portfolio Optimisation

Method 1: Mean - Variance MIQP model

Portf	8035	6701	6503	4685	6501	8036	4800	6256	9629	4273	1724	...	8793	7169	8584	8630
1	0.0	0.0	0.0	0.06	0.0	0.0	0.0	0.0	0.03	0.0	0.00	...	0.0	0.00	0.0	0.0
2	0.0	0.0	0.0	0.07	0.0	0.0	0.0	0.0	0.04	0.0	0.00	...	0.0	0.01	0.0	0.0
3	0.0	0.0	0.0	0.07	0.0	0.0	0.0	0.0	0.04	0.0	0.00	...	0.0	0.02	0.0	0.0
4	0.0	0.0	0.0	0.08	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.02	0.0	0.0
5	0.0	0.0	0.0	0.08	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.02	0.0	0.0
6	0.0	0.0	0.0	0.09	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.03	0.0	0.0
7	0.0	0.0	0.0	0.09	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.03	0.0	0.0
8	0.0	0.0	0.0	0.10	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.03	0.0	0.0
9	0.0	0.0	0.0	0.10	0.0	0.0	0.0	0.0	0.05	0.0	0.00	...	0.0	0.04	0.0	0.0
10	0.0	0.0	0.0	0.10	0.0	0.0	0.0	0.0	0.06	0.0	0.01	...	0.0	0.04	0.0	0.0
11	0.0	0.0	0.0	0.11	0.0	0.0	0.0	0.0	0.05	0.0	0.01	...	0.0	0.04	0.0	0.0
12	0.0	0.0	0.0	0.12	0.0	0.0	0.0	0.0	0.05	0.0	0.01	...	0.0	0.05	0.0	0.0
13	0.0	0.0	0.0	0.12	0.0	0.0	0.0	0.0	0.05	0.0	0.01	...	0.0	0.05	0.0	0.0
14	0.0	0.0	0.0	0.13	0.0	0.0	0.0	0.0	0.05	0.0	0.01	...	0.0	0.06	0.0	0.0
15	0.0	0.0	0.0	0.12	0.0	0.0	0.0	0.0	0.05	0.0	0.02	...	0.0	0.06	0.0	0.0
16	0.0	0.0	0.0	0.13	0.0	0.0	0.0	0.0	0.04	0.0	0.02	...	0.0	0.06	0.0	0.0
17	0.0	0.0	0.0	0.14	0.0	0.0	0.0	0.0	0.04	0.0	0.02	...	0.0	0.06	0.0	0.0
18	0.0	0.0	0.0	0.14	0.0	0.0	0.0	0.0	0.03	0.0	0.02	...	0.0	0.06	0.0	0.0
19	0.0	0.0	0.0	0.14	0.0	0.0	0.0	0.0	0.03	0.0	0.03	...	0.0	0.06	0.0	0.0
20	0.0	0.0	0.0	0.15	0.0	0.0	0.0	0.0	0.02	0.0	0.03	...	0.0	0.06	0.0	0.0
21	0.0	0.0	0.0	0.16	0.0	0.0	0.0	0.0	0.02	0.0	0.03	...	0.0	0.06	0.0	0.0
22	0.0	0.0	0.0	0.16	0.0	0.0	0.0	0.0	0.01	0.0	0.04	...	0.0	0.06	0.0	0.0
23	0.0	0.0	0.0	0.17	0.0	0.0	0.0	0.0	0.01	0.0	0.04	...	0.0	0.07	0.0	0.0
24	0.0	0.0	0.0	0.17	0.0	0.0	0.0	0.0	0.00	0.0	0.04	...	0.0	0.07	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
47	0.0	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.09	...	0.0	0.00	0.0	0.0
48	0.0	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.08	...	0.0	0.00	0.0	0.0
49	0.0	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.04	...	0.0	0.00	0.0	0.0
50	0.0	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.00	0.0	0.00	...	0.0	0.00	0.0	0.0

Table B.30: Pareto optimal portfolios for Tokyo securities

Method 2: Goal programming model

8035.T	6701.T	6503.T	4685.T	6501.T	8036.T	4800.T	6256.T	9629.T	6273.T
0.04	0.07	0.0	0.05	0.0	0.03	0.00	0.00	0.00	0.03
1724.T	7748.T	3671.T	3756.T	6920.T	6641.T	8057.T	6762.T	8031.T	8131.T
0.031	0.035	0.03	0.035	0.03	0.043	0.23	0.093	0.0	0.0
9818.T	5008.T	8002.T	7486.T	1514.T	3167.T	5021.T	6379.T	5020.T	2734.T
0.082	0.033	0.035	0.03	0.0	0.0	0.0	0.0	0.0	0.0
5015.T	2768.T	8037.T	8132.T	5019.T	8972.T	8976.T	3451.T	3453.T	8960.T
0.0	0.0	0.0	0.0	0.03	0.03	0.0	0.0	0.0	0.136
3772.T	3309.T	3455.T	8977.T	3308.T	8725.T	8793.T	7169.T	8584.T	8630.T
0.0	0.0	0.0	0.0	0.0	0.0	0.03	0.0	0.0	0.0

Table B.31: Goal programming portfolio for Tokyo stock exchange

Method 3: PROMETHEE flow multiobjective programming model

Portf	8035	6701	6503	4685	6501	8036	4800	6256	9629	4273	1724	...	8793	7169	8584	8630
1	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.00	0.03	0.00	0.03	...	0.00	0.03	0.00	0.00
2	0.00	0.00	0.00	0.03	0.00	0.00	0.03	0.00	0.03	0.00	0.03	...	0.03	0.00	0.00	0.00
3	0.00	0.03	0.00	0.03	0.00	0.03	0.00	0.00	0.03	0.00	0.03	...	0.03	0.00	0.00	0.00
4	0.00	0.03	0.00	0.03	0.00	0.03	0.00	0.03	0.03	0.00	0.03	...	0.03	0.03	0.00	0.00
5	0.03	0.03	0.00	0.03	0.00	0.03	0.00	0.03	0.03	0.03	0.03	...	0.03	0.00	0.00	0.00
6	0.03	0.03	0.03	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	...	0.03	0.00	0.00	0.00
7	0.03	0.03	0.03	0.03	0.03	0.03	0.00	0.03	0.03	0.00	0.00	...	0.03	0.00	0.00	0.00
8	0.03	0.03	0.03	0.03	0.03	0.05	0.00	0.03	0.03	0.00	0.00	...	0.03	0.00	0.00	0.00
9	0.03	0.03	0.03	0.03	0.03	0.09	0.00	0.03	0.03	0.00	0.00	...	0.03	0.00	0.00	0.00
10	0.03	0.03	0.03	0.03	0.03	0.12	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
11	0.03	0.03	0.03	0.03	0.03	0.18	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
12	0.03	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
13	0.07	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
14	0.10	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
15	0.13	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
16	0.16	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
17	0.20	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
18	0.20	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
19	0.20	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
20	0.16	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
21	0.10	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.00	0.00
22	0.12	0.03	0.03	0.00	0.03	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
23	0.05	0.03	0.03	0.00	0.07	0.20	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
24	0.03	0.03	0.03	0.00	0.14	0.15	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
25	0.09	0.03	0.03	0.00	0.20	0.03	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
26	0.08	0.03	0.03	0.00	0.20	0.04	0.00	0.03	0.00	0.03	0.00	...	0.03	0.03	0.03	0.03
27	0.08	0.03	0.04	0.00	0.20	0.03	0.00	0.03	0.00	0.03	0.00	...	0.03	0.03	0.03	0.03
28	0.08	0.03	0.03	0.00	0.20	0.04	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
29	0.06	0.03	0.03	0.00	0.20	0.06	0.00	0.03	0.00	0.03	0.00	...	0.03	0.00	0.03	0.03
30	0.03	0.00	0.03	0.00	0.20	0.03	0.00	0.03	0.00	0.03	0.00	...	0.00	0.00	0.03	0.03

Table B.32: Set of efficient portfolios for Tokyo stock exchange with MOIP PROMETHEE method

Method 4: Genetic algorithm model

8035.T	6701.T	6503.T	4685.T	6501.T	8036.T	4800.T	6256.T	9629.T	6273.T
0.0	0.03	0.0	0.03	0.0	0.03	0.03	0.03	0.03	0.03
1724.T	7748.T	3671.T	3756.T	6920.T	6641.T	8057.T	6762.T	8031.T	8131.T
0.031	0.03	0.03	0.2	0.03	0.0	0.03	0.193	0.0	0.0
9818.T	5008.T	8002.T	7486.T	1514.T	3167.T	5021.T	6379.T	5020.T	2734.T
0.082	0.038	0.0	0.0	0.0	0.03	0.0	0.0	0.0	0.0
5015.T	2768.T	8037.T	8132.T	5019.T	8972.T	8976.T	3451.T	3453.T	8960.T
0.0	0.0	0.0	0.0	0.03	0.0	0.0	0.0	0.0	0.036
3772.T	3309.T	3455.T	8977.T	3308.T	8725.T	8793.T	7169.T	8584.T	8630.T
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.03

Table B.33: Genetic algorithm portfolio for Tokyo stock exchange

Bibliography

[1] Xidonas P. et al. *Multicriteria Portfolio Management*. Springer, 2012.

[2] Roy B. *Classement et choix en presence de points de vue multiples (la methode ELECTRE).Revue Informatique et Recherche Operationnelle 8, 57-75,* 1968.

[3] Roy B. *Methodologie multicritere d'aide a la Decision, Economica, Paris.,* 1985.

[4] Roy B. *Multicriteria methodology for decision aiding. Kluwer Academic Publishers, Dordrecht,* 1996.

[5] Roy B. *The outranking approach and the foundations of ELECTRE methods. Theory and Decision, 31, 49-73,* 1991.

[6] Hunter J. D. "Matplotlib: A 2D graphics environment". In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[7] Oliphant Travis E. *A guide to NumPy*. Vol. 1. Trelgol Publishing USA, 2006.

[8] Dantzig G. "Linear programming and extensions. Princeton, New Jersey, Princeton University Press," in: (1998).

[9] Arrow K. Raynaud H. *Social choice and multicriterion decision-making. Cambridge, Massachusetts, MIT Press.,* 1986.

[10] Keeney R.L. Raiffa H. *Decisions with multiple objectives: Preferences and value tradeoffs. Cambridge University Press, Cambridge,* 1993.

[11] Markowitz H. "Portfolio selection". In: (1952), pp. 77–91.

[12] Markowitz H. "Portfolio Selection: Efficient diversification of investments". In: (1959).

[13] Jacquet-Lagreze E. Siskos J. *Assessing a set of additive utility functions for multicriteria decision-making, the UTA method. European Journal of Operational Research, 10 (2), 151-164,* 1982.

[14] Jacquet-Lagreze E. Siskos J. *Preference disaggregation: Twenty years of MCDA experience. European Journal of Operational Research, 130 (2), 233-245,* 2001.

[15] Xidonas P. Mavrotas G. Zopounidis C. Psarras J. *IPSSIS: An integrated multicriteria decision support system for equity portfolio construction and selection.* 2012.

[16] Hwang C.L. Yoon K. *Multiple attribute decision making: Methods and applications. Springer Verlag, New York,* 1981.

[17] Wes McKinney. "Data Structures for Statistical Computing in Python". In: *Proceedings of the 9th Python in Science Conference*. Ed. by Stéfan van der Walt and Jarrod Millman. 2010, pp. 51–56.

[18] Steuer R. *Multiple criteria optimization: Theory, computation and applications. New York, John Wiley and Sons,* 1986.

[19] Charnes A. Cooper W.W. Ferguson R.O. *Optimal estimation of executive compensation by linear programming. Management Science, 1 (2), 138-151,* 1955.

[20] Benninga S. *Financial Modeling*. 3rd Edition. MIT Press, 2008.

[21] Charnes A. Cooper W.W. *Management models and industrial applications of linear programming. Wiley, New York,* 1961.

[22] Mavrotas G. Xidonas P. *Multiobjective portfolio optimization with non-convex policy constraints: Evidence from the Eurostoxx 50, The European Journal of Finance, DOI:10.1080/1351847X.2012.733718,* 2012.

[23] Ξυδώνας Π. Ψαρράς Ι. Ζοπουνίδης Κ. *Διαχείριση χαρτοφυλακίων με πολλαπλά κριτήρια: Θεωρία και πράξη*. Αθήνα: Εκδόσεις Κλειδάριθμος, 2010.

[24] Ξυδώνας Π. Ψαρράς Ι. Ζοπουνίδης Κ. *Σύγχρονη θεωρία χαρτοφυλακίου*. Αθήνα: Εκδόσεις Κλειδάριθμος, 2010.

[25] Δούμπος Μ. *Μαθηματικός Χρηματοοικονομικός Λογισμός*. 2010.

[26] Δούμπος Μ. *Πολυκριτήρια συστήματα αποφάσεων. Σημειώσεις μαθήματος. Πολυτεχνείο Κρήτης, Χανιά,* 2009.

[27] Ξυδώνας Π. "Ολοκληρωμένη Μεθοδολογία & Πληροφοριακό Σύστημα Υποστήριξης Αποφάσεων για τη Διαχείριση Μετοχικών Χαρτοφυλακίων". Διδακτορική Διατριβή. Αθήνα, 2010.