



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΙΩΑΝΝΙΝΩΝ

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΧΡΟΝΟΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΠΑΡΟΥΣΙΑΣΗΣ
ΔΙΠΛΩΜΑΤΙΚΩΝ ΕΡΓΑΣΙΩΝ

Ευθαλία Παππά

Επιβλέπων: Χρήστος Γκόγκος
Αναπληρωτής καθηγητής Πανεπιστημίου Ιωαννίνων

Δεκεμβρίου 2022

THESIS DEFENSE TIMETABLING

Εγκρίθηκε από τριμελή εξεταστική επιτροπή

Τόπος, Ημερομηνία

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Επιβλέπων καθηγητής

Όνομα Επίθετο,

τίτλος, βαθμίδα

2. Μέλος επιτροπής

Όνομα Επίθετο,

τίτλος, βαθμίδα

3. Μέλος επιτροπής

Όνομα Επίθετο,

τίτλος, βαθμίδα

© Παππά, Ευθαλία, 2022.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Δήλωση μη λογοκλοπής

Δηλώνω υπεύθυνα και γνωρίζοντας τις κυρώσεις του Ν. 2121/1993 περί Πνευματικής Ιδιοκτησίας, ότι η παρούσα μεταπτυχιακή εργασία είναι εκ ολοκλήρου αποτέλεσμα δικής μου ερευνητικής εργασίας, δεν αποτελεί προϊόν αντιγραφής ούτε προέρχεται από ανάθεση σε τρίτους. Όλες οι πηγές που χρησιμοποιήθηκαν (κάθε είδους, μορφής και προέλευσης) για τη συγγραφή της περιλαμβάνονται στη βιβλιογραφία.

Ευθαλία Παππά

Υπογραφή

ΕΥΧΑΡΙΣΤΙΕΣ

Για τη διεκπεραίωση της παρούσας Διπλωματικής Εργασίας, θα ήθελα να ευχαριστήσω αρχικά τον επιβλέποντα καθηγητή μου, κ. Χρήστο Γκόγκο για την ευκαιρία που μου έδωσε να ανακαλύψω το πεδίο του χρονοπρογραμματισμού, καθώς και τον κ. Άγγελο Δήμητσα για την συμβολή του στην ολοκλήρωση της διπλωματικής και για τη συνεχή βοήθεια και υποστήριξη σε όσα προβλήματα παρουσιάστηκαν. Τέλος, θέλω να ευχαριστήσω την οικογένειά μου για την αμέριστη υποστήριξη και υπομονή τους για όλη την διάρκεια των μεταπτυχιακών σπουδών μου.

ΠΕΡΙΛΗΨΗ

Ένα σημαντικό καθήκον της διοίκησης κάθε πανεπιστημίου είναι ο προγραμματισμός της παρουσίας των διπλωματικών εργασιών και η τελετή αποφοίτησης. Οι υποψήφιοι φοιτητές είναι υποχρεωμένοι να υπερασπιστούν την διατριβή τους μπροστά σε μία επιτροπή που απαρτίζεται από μέλη ΔΕΠ του εκάστοτε ιδρύματος. Τα μεγάλα ευρωπαϊκά τμήματα, όμως, μπορεί να έχουν πολλούς φοιτητές που αποφοιτούν την ίδια χρονική περίοδο, με αποτέλεσμα να είναι αναγκαίο να προγραμματιστούν πολλές συνεδρίες με διαφορετικές επιτροπές και πιθανόν σε διαφορετικές ημέρες. Ο συγκεκριμένος τρόπος υπεράσπισης της διατριβής παρατηρείται κυρίως στα μεγάλα ιδρύματα της Ευρώπης και όχι στα ελληνικά πανεπιστήμια.

Για πολλές δεκαετίες η επίλυση του χρονοπρογραμματισμού και η παραγωγή βέλτιστης λύσης θεωρείται ένα δύσκολο πρόβλημα, διότι η παραγωγή χειροκίνητων προγραμμάτων απαιτεί πολύ χρόνο και κόπο. Το πρόβλημα που θα μελετηθεί στην παρούσα εργασία αναφέρεται στον *χρονοπρογραμματισμό* των συνεδριών για την υπεράσπιση της διατριβής, αγγλική ορολογία: *Thesis Defense TimeTabling* (TDDT) problem, όπου συνίσταται τόσο στην ανάθεση κάθε υποψηφίου σε μία συνεδρία όσο και στη σύνθεση των κατάλληλων επιτροπών, ικανοποιώντας διάφορους περιορισμούς και στόχους. Το πρόβλημα αυτό είναι καινούριο στη βιβλιογραφία, και έχει μελετηθεί λίγο στο πεδίο του χρονοπρογραμματισμού.

Έχουν προταθεί διαφορετικοί τρόποι μοντελοποίησης για το πρόβλημα TDDT από αρκετούς ερευνητές στο παρελθόν. Εδώ, μελετάμε μια συγκεκριμένη εκδοχή του προβλήματος που προτάθηκε στο *Modelling and solving the thesis defense timetabling problem* [1], και παρέχουμε τρεις βελτιωμένες μεθόδους λύσης, που βασίζονται στο Μικτό Ακέραιο Προγραμματισμό (MIP) με χρήση της CPLEX, στον Προγραμματισμό με Περιορισμούς και στην Τοπική Αναζήτηση.

Λέξεις-κλειδιά: Χρονοπρογραμματισμός, Μικτός Ακέραιος Προγραμματισμός, Τοπική Αναζήτηση

ABSTRACT

An important task carried out by the administration of each university is the scheduling of the thesis defense timetabling and the graduation ceremony. Candidate students are required to defend their thesis in front of a committee composed of faculty members of the respective institution. Large European departments, however, may have many students graduating at the same time, making it necessary to schedule several sessions with different committees and possibly on different days. This particular way of defending the thesis is mainly observed in large European institutions and not in Greek universities.

For many decades, solving scheduling and generating an optimal solution has been considered a difficult challenge, as generating manual schedules requires a lot of time and effort. The problem that is studied in the present thesis refers to the scheduling of thesis defense sessions, Thesis Defense Timetabling (TDDT) problem, which comprises both the assignment of each candidate to a session and the composition of appropriate committees, satisfying various constraints and objectives. Being new in the literature, this problem has been little studied in the field of scheduling.

Different modeling approaches for the TDDT problem have been proposed by several researchers in the past. In this thesis, we reflect on a particular version of the problem proposed in Modelling and Solving the Thesis Defense Timetabling Problem and suggest three improved solution methods based on Mixed Integer Programming (MIP) using CPLEX, Constraint Programming (CPLEX) and Local Search .

Keywords: Timetabling; integer programming; local search; simulated annealing; constraint programming

Περιεχόμενα

ΕΥΧΑΡΙΣΤΙΕΣ	6
ΠΕΡΙΛΗΨΗ	7
ABSTRACT	8
ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ	11
Κατάλογος Εικόνων	12
Κεφάλαιο 1	13
1.1 Διατύπωση Προβλήματος	13
1.2 Μοντελοποίηση	15
1.3 Περιγραφή Δεδομένων	17
1.3.1 Σύνολο Δεδομένων	17
1.3.2 Ανάλυση Δεδομένων	19
Κεφάλαιο 2	28
2.1 Χρονοπρογραμματισμός.....	28
2.1.1 Προγραμματισμός εκπαιδευτικών ιδρυμάτων.....	29
2.1.2 Προβλήματα προγραμματισμού εκπαιδευτικών ιδρυμάτων	29
2.1.3 Σχετικές εργασίες	30
2.2 Μαθηματικός Προγραμματισμός	31
2.2.1 Γραμμικός Προγραμματισμός.....	32
2.2.2 Ακέραιος Προγραμματισμός	33
2.2.3 Μικτός Ακέραιος Προγραμματισμός	34
2.3 Προγραμματισμός υπό περιορισμούς.....	35
2.4 Ευρετικοί Αλγόριθμοι	36
2.5 Μεταευρετικοί Αλγόριθμοι	37
2.6 Προσομοιωμένη Ανόπτηση.....	38
2.7 Γενετικός Αλγόριθμος.....	39
2.8 Βελτιστοποίηση Σμήνους Σωματιδίων.....	39
2.9 Υπολογιστική Πολυπλοκότητα	40
Κεφάλαιο 3	41
3.1 Πειράματα	41
Μεταευρετική μέθοδος.....	41
3.2 Μικτός Ακέραιος Προγραμματισμός	48
3.4 Αποτελέσματα	52
Κεφάλαιο 4.....	58
Συμπέρασμα	58
Βιβλιογραφία	59

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Τα κύρια χαρακτηριστικά για κάθε στιγμιότυπο	17
Πίνακας 2: Συγκεντρωτικός πίνακας στατιστικών στοιχείων των δεδομένων	26
Πίνακας 3: Συνέχεια Πίνακα 2.....	27
Πίνακας 4: Ταξινομημένη λίστα κωδικών της κατάστασης που παρέχεται από τη CPLEX	34
Πίνακας 5: Τιμές κατάστασης που εμφανίζει το πρόγραμμα επίλυσης CP-SAT	35
Πίνακας 6: Αφαιρέστε έναν καθηγητή από μία συνεδρία.....	43
Πίνακας 7: Αναθέστε σε διαφορετική συνεδρία μια ομάδα φοιτητών (μαζί με τον επόπτη) που έχουν ανατεθεί στην ίδια συνεδρία και έχουν τον ίδιο επόπτη	44
Πίνακας 8: Αναθέστε έναν καθηγητή σε μία τυχαία συνεδρία (στην οποία είναι διαθέσιμος)	45
Πίνακας 9: Αναθέστε έναν φοιτητή σε διαφορετική συνεδρία και προσθέστε τον επιβλέποντά του και στην καινούρια συνεδρία (παραμένει στην παλιά συνεδρία)	46
Πίνακας 10: Ανταλλάξτε συνεδρίες δύο ομάδων φοιτητών (μαζί και τους αντίστοιχους επόπτες), όπου οι φοιτητές κάθε ομάδας βρίσκονται στην ίδια συνεδρία και έχουν τον ίδιο επόπτη	47
Πίνακας 11: Αποτελέσματα της εφαρμογής τριών φάσεων στις πραγματικές περιπτώσεις για CP Sat και CPLEX.....	53
Πίνακας 12: Αποτελέσματα της εφαρμογής τριών φάσεων στις τεχνητές περιπτώσεις για CP Sat και CPLEX.....	54
Πίνακας 13: Αποτελέσματα εφαρμογής με Προσομοιωμένη Ανόπτηση για τις πραγματικές περιπτώσεις.....	55
Πίνακας 14: Αποτελέσματα εφαρμογής με Προσομοιωμένη Ανόπτηση για τις τεχνητές περιπτώσεις.....	57

Κατάλογος Εικόνων

Εικόνα 1	19
Εικόνα 2	19
Εικόνα 3	20
Εικόνα 4	20
Εικόνα 5	21
Εικόνα 6	22
Εικόνα 7	22
Εικόνα 8	23
Εικόνα 9	23
Εικόνα 10	24
Εικόνα 11	24
Εικόνα 12	25
Εικόνα 13	30
Εικόνα 14	40

Κεφάλαιο 1

1.1 Διατύπωση Προβλήματος

Το πρόβλημα που μελετάμε αφορά τον χρονοπρογραμματισμό της υπεράσπισης της διατριβής αποφοίτων σε διαφορετικές συνεδρίες και με την κατάλληλη επιτροπή, όπου κάθε υποψήφιος φοιτητής εκχωρείται σε μία μοναδική συνεδρία και διαθέτει ένα σύνολο μελών ΔΕΠ που είναι κατάλληλοι ως αντίπαλοί του και έναν μοναδικό επιβλέποντα. Η κάθε συνεδρία αποτελείται από ένα μέγιστο αριθμό υποψηφίων, και στην περίπτωση που οι συνεδρίες δεν επαρκούν για όλους τους φοιτητές προστίθενται καινούριες, παράλληλα με αυτές που έχουν ήδη προγραμματιστεί. Σύμφωνα με την επιτροπή, τα μέλη ΔΕΠ μπορούν να συμμετέχουν σε περισσότερες από μία επιτροπές (ή σε καμία). Το πλήθος των μελών σε μία συνεδρία δεν είναι σταθερό, αλλά καθορίζεται από τα όρια κάθε ακαδημαϊκού επιπέδου.

Συνοψίζοντας, τα κύρια μέρη του προβλήματος είναι:

- **Συνεδρίες:** Συναντήσεις στις οποίες οι φοιτητές υπερασπίζονται τη διατριβή τους μπροστά σε μία επιτροπή (μερικές συνεδρίες μπορεί να επικαλύπτονται).
- **Μέλη ΔΕΠ:** Ο κατάλογος του πανεπιστημιακού προσωπικού που μπορούν να διοριστούν ως μέλη επιτροπής. Χαρακτηρίζονται από το ακαδημαϊκό τους επίπεδο και τη λίστα των συνεδριών στις οποίες δεν μπορούν να παρακολουθήσουν.
- **Φοιτητές:** Ο κατάλογος των υποψηφίων αποφοίτησης. Για κάθε φοιτητή, δίνεται το μέλος ΔΕΠ που τον επιβλέπει και μία λίστα των πιθανών αντιπάλων. Επιβλέπων και αντίπαλοι είναι μέλη της σχολής.

Διατύπωση Περιορισμών: Οι περιορισμοί συνήθως χωρίζονται σε αυστηρούς και χαλαρούς. Ένας αυστηρός περιορισμός καθιστά ένα χρονοδιάγραμμα ανέφικτο, ενώ οι χαλαροί περιορισμοί επιβάλλουν ποινή στην αντικειμενική συνάρτηση εάν δεν ικανοποιούνται. Στόχος μας είναι να ικανοποιήσουμε όλους τους αυστηρούς περιορισμούς ελαχιστοποιώντας παράλληλα τις παραβιάσεις των χαλαρών περιορισμών.

Οι *αυστηροί* περιορισμοί είναι:

- **H1. Επίβλεψη:** Για κάθε φοιτητή, ο επιβλέπων καθηγητής πρέπει να είναι παρών στη συνεδρία που έχει ανατεθεί στον φοιτητή.
- **H2. Φοιτητές ανά συνεδρία:** Ο αριθμός των φοιτητών που αναθέτονται σε μια συνεδρία δεν πρέπει να υπερβαίνει ένα δεδομένο μέγιστο.
- **H3. Επικάλυψη συνεδριών:** Σε περίπτωση που δύο συνεδρίες γίνονται την ίδια στιγμή, ένα μέλος της σχολής μπορεί να ανατεθεί το πολύ σε μία από τις δύο επιτροπές.
- **H4. Διαθεσιμότητα:** Κάθε μέλος της επιτροπής πρέπει να είναι διαθέσιμο στη συνεδρία που έχει προγραμματιστεί.
- **H5. Σύνθεση επιτροπής:** Οι συνεδρίες πρέπει να σχηματίζονται τηρώντας έναν ελάχιστο και μέγιστο αριθμό προσωπικού ΔΕΠ που κατέχει συγκεκριμένη ακαδημαϊκή βαθμίδα. Το προσωπικό με υψηλότερη ακαδημαϊκή βαθμίδα

μπορεί να αντικαταστήσει το προσωπικό με χαμηλότερη βαθμίδα αλλά όχι το αντίστροφο.

Για παράδειγμα, εάν σε μία περίπτωση προβλήματος υπάρχουν τέσσερις βαθμίδες, οι οποίες είναι (ξεκινώντας από την υψηλότερη): Τακτικός(Ομότιμος) Καθηγητής, Αναπληρωτής Καθηγητής, Επίκουρος Καθηγητής και Εξωτερικός Καθηγητής (ή Επιστημονικός Συνεργάτης) και απαιτούνται τουλάχιστον τρεις επίκουροι καθηγητές, τότε αυτός ο περιορισμός μπορεί επίσης να εκπληρωθεί με δύο αναπληρωτές καθηγητές και έναν επίκουρο καθηγητή. Κατά συνέπεια, τα όρια για τη χαμηλότερη ακαδημαϊκή βαθμίδα αντιπροσωπεύουν τα όρια του συνολικού αριθμού μελών, καθώς όλα τα μέλη ΔΕΠ έχουν το χαμηλότερη βαθμίδα (ή μία υψηλότερη βαθμίδα).

Οι χαλαροί περιορισμοί είναι:

S1. Πολλαπλά Καθήκοντα: Κάθε φορά που ένα μέλος ΔΕΠ συμμετέχει σε περισσότερες από μία επιτροπές, επιβάλλεται ποινή. Για την αποφυγή μεγάλου φόρτου εργασίας, η ποινή της παράβασης είναι τετραγωνική. Δηλαδή, εάν α είναι ο αριθμός των παρουσιών ενός μέλους ΔΕΠ, η σχετική ποινή είναι $(\alpha-1)^2$ για $\alpha > 0$, και 0 για $\alpha = 0$.

S2. Παρουσία αντιπάλου: Επιβάλλεται ποινή εάν ένας υποψήφιος παρευρεθεί σε μια συνεδρία χωρίς να είναι παρών κανένας από τους κατάλληλους αντιπάλους του. Εάν ο υποψήφιος δεν έχει κατάλληλους αντιπάλους δεν επιβάλλεται ποινή.

Η αντικειμενική συνάρτηση προκύπτει αθροίζοντας τις παραβιάσεις των δύο χαλαρών περιορισμών, ο καθένας πολλαπλασιαζόμενος με το αντίστοιχο βάρος. Τα βάρη που επιλέχθηκαν σε αυτήν την εργασία είναι: $w_{S1} = 1$ και $w_{S2} = 3$ για τους χαλαρούς περιορισμούς S1 και S2 αντίστοιχα.

1.2 Μοντελοποίηση

Για την επίλυση του προβλήματος TDDT ορίζουμε ένα μοντέλο Μικτού Ακέραιου Τετραγωνικού Προγραμματισμού (MIQP). Αρχικά ορίζουμε τα δεδομένα εισόδου, έπειτα τις μεταβλητές απόφασης και τέλος το μαθηματικό μοντέλο.

Τα δεδομένα εισόδου περιγράφονται παρακάτω:

Έστω

- $C = \{1, \dots, C\}$: Το σύνολο των υποψηφίων αποφοίτησης
- $F = \{1, \dots, F\}$: Το σύνολο των μελών της επιτροπής
- F_c : Το μέλος επιτροπής που επιβλέπει τον c υποψήφιο.
- $S = \{1, \dots, S\}$: Το σύνολο των συνεδριών
- O : Το σύνολο των ζευγών από επικαλυπτόμενες συνεδρίες
- S_f : Το σύνολο των συνεδριών που ένα μέλος δεν είναι διαθέσιμο
- Q_c : Το σύνολο των κατάλληλων αντιπάλων ενός υποψηφίου c
- $L = \{1, \dots, L\}$: Το σύνολο των ακαδημαϊκών επιπέδων
- C : ο μέγιστος αριθμός υποψηφίων σε μια συνεδρία
- m_l και M_l : ελάχιστος και μέγιστος απαιτούμενος αριθμός μελών ΔΕΠ ανά συνεδρία για ακαδημαϊκή βαθμίδα l
- A_f : Το ακαδημαϊκό επίπεδο ενός μέλους επιτροπής f

Επιπρόσθετα, ορίζουμε τις ακόλουθες μεταβλητές απόφασης που αφορούν την ανάθεση φοιτητών και μελών ΔΕΠ σε συνεδρίες:

Μεταβλητές Απόφασης

$$x_{cs} = \begin{cases} 1, & \text{αν ο υποψήφιος } c \text{ ανατίθεται σε μία συνεδρία } s. \\ 0, & \text{αλλιώς.} \end{cases} \quad \forall c \in C, \forall s \in S \quad (1)$$

$$y_{fs} = \begin{cases} 1, & \text{αν το μέλος επιτροπής } f \text{ ανατίθεται σε μία συνεδρία } s. \\ 0, & \text{αλλιώς.} \end{cases} \quad \forall f \in F, \forall s \in S \quad (2)$$

Ορίζουμε επίσης τις ακόλουθες δευτερεύουσες μεταβλητές απόφασης που χρησιμοποιούνται στην αντικειμενική συνάρτηση.

$$z_c = \begin{cases} 1, & \text{εάν ο υποψήφιος δεν έχει έναν κατάλληλο αντίπαλο.} \\ 0, & \text{αλλιώς.} \end{cases} \quad \forall c \in C, |Q_c| \neq 0 \quad (3)$$

$$u_f \in [0, S - 1] \quad (4)$$

Επομένως, το μοντέλο είναι το ακόλουθο:

Αντικειμενική Συνάρτηση:

$$\min w_{s1} * \sum_{f \in \mathbb{F}} u_f^2 + w_{s2} * \sum_{c \in \mathbb{C}, |\mathbb{Q}_c| \neq 0} z_c \quad (5)$$

Περιορισμοί:

$$\sum_{s \in \mathbb{S}} x_{cs} = 1 \quad \forall c \in \mathbb{C} \quad (6)$$

$$x_{cs} \leq y_{fcs} \quad \forall c \in \mathbb{C}, \forall s \in \mathbb{S} \quad (7)$$

$$\sum_{c \in \mathbb{C}} x_{cs} \leq C \quad \forall s \in \mathbb{S} \quad (8)$$

$$y_{fs_1} + y_{fs_2} \leq 1 \quad \forall s_1, s_2 \in \mathbb{Q}, \forall f \in \mathbb{F} \quad (9)$$

$$y_{fs} = 0 \quad \forall f \in \mathbb{F}, \forall s \in \mathbb{S}_f \quad (10)$$

$$m_l \leq \sum_{f \in \mathbb{F}, A_f \leq 1} y_{fs} \leq M_l \quad \forall l \in \mathbb{L}, \forall s \in \mathbb{S} \quad (11)$$

$$\sum_{f \in \mathbb{Q}_c} (y_{fs}) + z_c \geq x_{cs} \quad \forall c \in \mathbb{C}, |\mathbb{Q}_c| \neq 0, \forall s \in \mathbb{S} \quad (12)$$

$$\sum_{s \in \mathbb{S}} (y_{fs}) - 1 \leq u_f \quad \forall f \in \mathbb{F} \quad (13)$$

Ο περιορισμός 6 επιβάλλει ότι ένας υποψήφιος παρακολουθεί ακριβώς μια συνεδρία και ο περιορισμός 7 ότι ο επιβλέπωντάς του θα είναι επίσης παρών στην υπεράσπιση του υποψηφίου. Ο περιορισμός 8 περιορίζει τον αριθμό των υποψηφίων σε κάθε συνεδρία σύμφωνα με τη μέγιστη επιτρεπόμενη χωρητικότητα. Ο περιορισμός 9 διασφαλίζει ότι κανένα μέλος ΔΕΠ δεν έχει προγραμματιστεί σε επικαλυπτόμενες συνεδρίες.

Η εξίσωση 10 αναγκάζει ένα μέλος επιτροπής να μην εμφανίζεται σε συνεδρίες που δεν είναι διαθέσιμο. Οι απαιτήσεις της επιτροπής του αυστηρού περιορισμού Η4 τηρούνται βάσει του περιορισμού 11. Τέλος, ο περιορισμός 12 διασφαλίζει ότι η μεταβλητή απόφασης z θα ενεργοποιηθεί εάν ο υποψήφιος δεν έχει τοποθετηθεί με τουλάχιστον έναν κατάλληλο αντίπαλο και ο περιορισμός 13 επιβάλλει την ποινή του τετραγωνικού μέρους της αντικειμενικής συνάρτησης.

1.3 Περιγραφή Δεδομένων

1.3.1 Σύνολο Δεδομένων

Σε αυτήν την ενότητα περιγράφονται τα δεδομένα που χρησιμοποιήθηκαν για την δοκιμή του μοντέλου του TDDT. Μαζί με τον ορισμό του προβλήματος δημοσιοποιήθηκε ένα σύνολο δεδομένων, το οποίο αποτελείται από 21 πραγματικά στιγμιότυπα που λαμβάνονται από Ιταλικά Πανεπιστήμια, όπως το Πανεπιστήμιο της Bicocca (Μιλάνο), το Πανεπιστήμιο του Μιλάνου και το Πανεπιστήμιο των Μοντένα και Reggio Emilia, καθώς και 45 ακόμη στιγμιότυπα που έχουν δημιουργηθεί τεχνητά έτσι ώστε να παρουσιάζουν παρόμοια χαρακτηριστικά με τα πραγματικά στιγμιότυπα (με Ανάλυση κυρίων συνιστωσών). Στον Πίνακα 1 παρουσιάζονται τα κύρια χαρακτηριστικά των στιγμιότυπων για κάθε περίπτωση προβλήματος, όπως ο αριθμός υποψηφίων, ο αριθμός μελών επιτροπής, η μέγιστη χωρητικότητα των συνεδριών και ο αριθμός συνεδριών. Περαιτέρω χαρακτηριστικά για κάθε περίπτωση προβλήματος, όπως ο αριθμός των ζευγών των επικαλυπτόμενων συνεδριών, συνεδρίες που δεν είναι διαθέσιμο κάθε μέλος επιτροπής, πιθανοί αντίπαλοι για κάθε φοιτητή και άλλα, μπορείτε να δείτε στη διεύθυνση <https://opthub.uniud.it/>. Επίσης, εξήγαμε κάποια επιπλέον χαρακτηριστικά για κάθε στιγμιότυπο.

Στιγμιότυπα	Υποψήφιοι	Μέλη Επιτροπής	Μέγιστο- υποψηφίων	Συνεδρίες	Στιγμιότυπα	Υποψήφιοι	Μέλη Επιτροπής	Μέγιστο- υποψηφίων	Συνεδρίες
tdtt-real01.dzn	143	155	9	16	tdtt-art13.dzn	110	52	10	13
tdtt-real02.dzn	34	79	10	5	tdtt-art14.dzn	459	504	9	55
tdtt-real03.dzn	144	77	8	18	tdtt-art15.dzn	522	290	10	56
tdtt-real04.dzn	171	101	10	18	tdtt-art16.dzn	532	461	10	60
tdtt-real05.dzn	442	296	12	37	tdtt-art17.dzn	532	796	10	60
tdtt-real06.dzn	79	99	8	10	tdtt-art18.dzn	249	413	10	28
tdtt-real07.dzn	102	66	13	8	tdtt-art19.dzn	183	274	10	19
tdtt-real08.dzn	55	67	10	6	tdtt-art20.dzn	555	671	10	57
tdtt-real09.dzn	173	280	14	15	tdtt-art21.dzn	457	724	10	51
tdtt-real10.dzn	209	435	7	31	tdtt-art22.dzn	276	405	8	37
tdtt-real11.dzn	164	287	12	16	tdtt-art23.dzn	269	127	9	31
tdtt-real12.dzn	222	322	14	16	tdtt-art24.dzn	42	62	10	5
tdtt-real13.dzn	430	435	13	35	tdtt-art25.dzn	71	37	8	9
tdtt-real14.dzn	551	454	13	45	tdtt-art26.dzn	169	187	8	23
tdtt-real15.dzn	428	454	7	66	tdtt-art27.dzn	576	277	10	59
tdtt-real16.dzn	124	84	10	13	tdtt-art28.dzn	326	441	10	34
tdtt-real17.dzn	18	76	10	2	tdtt-art29.dzn	366	196	8	48
tdtt-real18.dzn	172	80	11	18	tdtt-art30.dzn	361	617	10	42
tdtt-real19.dzn	69	79	11	7	tdtt-art31.dzn	263	351	11	26
tdtt-real20.dzn	146	81	11	14	tdtt-art32.dzn	73	192	10	8
tdtt-real21.dzn	58	31	15	4	tdtt-art33.dzn	134	240	10	14
tdtt-art01.dzn	307	251	8	41	tdtt-art34.dzn	52	172	10	6
tdtt-art02.dzn	40	30	8	6	tdtt-art35.dzn	117	205	10	12
tdtt-art03.dzn	208	240	9	24	tdtt-art36.dzn	112	204	10	12
tdtt-art04.dzn	347	409	9	43	tdtt-art37.dzn	90	207	10	10
tdtt-art05.dzn	137	261	8	18	tdtt-art38.dzn	176	296	11	18
tdtt-art06.dzn	434	267	10	45	tdtt-art39.dzn	282	360	11	28
tdtt-art07.dzn	467	261	9	54	tdtt-art40.dzn	243	339	11	24
tdtt-art08.dzn	427	705	9	54	tdtt-art41.dzn	104	207	10	11
tdtt-art09.dzn	73	64	10	8	tdtt-art42.dzn	164	290	11	17
tdtt-art10.dzn	40	39	8	6	tdtt-art43.dzn	141	269	11	15
tdtt-art11.dzn	395	209	9	49	tdtt-art44.dzn	89	227	10	10
tdtt-art12.dzn	255	420	9	32	tdtt-art45.dzn	225	330	11	23

Πίνακας 1: Τα κύρια χαρακτηριστικά για κάθε στιγμιότυπο

Παράδειγμα στιγμιότυπου (instance tdt-art02)

Sessions = 6; (αριθμός συνεδριών αποφοίτησης)

FacultyMembers = 30; (αριθμός μελών ΔΕΠ)

Candidates = 40; (αριθμός φοιτητών)

MaxCandidates = 8; (μέγιστος αριθμός φοιτητών ανά συνεδρία)

Levels = 4; (Ακαδημαϊκές βαθμίδες: 1= Ομότιμος Καθηγητής, 2= Αναπληρωτής Καθηγητής, 3= Επίκουρος Καθηγητής, 4= Εξωτερικός Καθηγητής)

LimitMembersForLevel = [|1, 8|3, 8|7, 8|7, 10|]; [|min, max|,...] : Ελάχιστο και μέγιστο αριθμό καθηγητών ανά βαθμίδα (ή υψηλότερη βαθμίδα), π.χ. Στο |1, 8| έχουμε ελάχιστο 1 Ομότ. Καθηγητών και μέγιστο 8 Ομότ. Καθηγητών

SimultaneityPairs = 1; (αριθμός από ζεύγη συνεδριών που γίνονται ταυτόχρονα)

SimultaneousSessions = [|1, 3|]; (η συνεδρία 1 και 3 γίνονται την ίδια στιγμή)

AcademicLevel = [3, 2, 1, 2, 4, 1, 4, 2, 1, 3, 3, 2, 4, 1, 1, 1, 4, 3, 1, 3, 1, 3, 3, 1, 2, 2, 3, 3, 3, 2]; (ακαδημαϊκή βαθμίδα για καθένα από τους 30 καθηγητές)

Unavailability = [{1, 2, 5}, {}, {4, 6}, {3, 4}, {1, 2}, {2, 5}, {3, 4, 5}, {1, 3, 4, 5}, {1, 3}, {2}, {1, 2, 4}, {5, 6}, {3, 4, 5, 6}, {5, 6}, {2, 3, 4}, {}, {3}, {6}, {3}, {1, 6}, {2, 6}, {3, 4, 5}, {2, 4}, {1, 2, 3, 4, 5}, {3, 4}, {4}, {1, 3, 6}, {1, 3}, {2, 4, 6}, {1, 3, 6}]; (συνεδρίες που δεν είναι διαθέσιμος κάθε καθηγητής)

Supervisor = [23, 24, 9, 27, 3, 24, 1, 28, 12, 6, 4, 6, 8, 9, 15, 30, 21, 25, 2, 22, 19, 12, 22, 28, 25, 29, 29, 6, 17, 2, 25, 2, 30, 10, 14, 11, 1, 23, 5, 22]; (ο επιβλέπωντας του κάθε φοιτητή)

Opponents = [{14, 16}, {5, 18}, {4, 16, 23}, {26, 29, 30}, {1, 26, 27}, {5, 18}, {3, 30}, {11, 15, 19}, {20, 25}, {8, 16}, {9, 16}, {2, 16}, {2, 6}, {10, 14}, {7, 11, 30}, {3, 29}, {5, 26}, {17, 20, 22}, {6, 8}, {17, 20, 25}, {1, 15}, {17, 25}, {20, 25}, {1, 11, 15}, {17, 20, 22}, {7, 27}, {7, 30}, {2, 8, 16}, {22, 25}, {6, 8}, {17, 22}, {6, 8}, {3, 19, 28}, {9, 16}, {10, 20}, {19, 28}, {3, 15, 28}, {9, 10}, {21, 24}, {12, 17, 20}]; (πιθανοί αντίπαλοι για κάθε φοιτητή, η κενή λίστα σημαίνει ότι καθένας μπορεί να είναι αντίπαλος)

1.3.2 Ανάλυση Δεδομένων

Σύμφωνα με το παραπάνω σύνολο δεδομένων βρέθηκαν κάποια στατιστικά στοιχεία για την κάθε περίπτωση στιγμιότυπου για την καλύτερη κατανόηση των δεδομένων του προβλήματος.

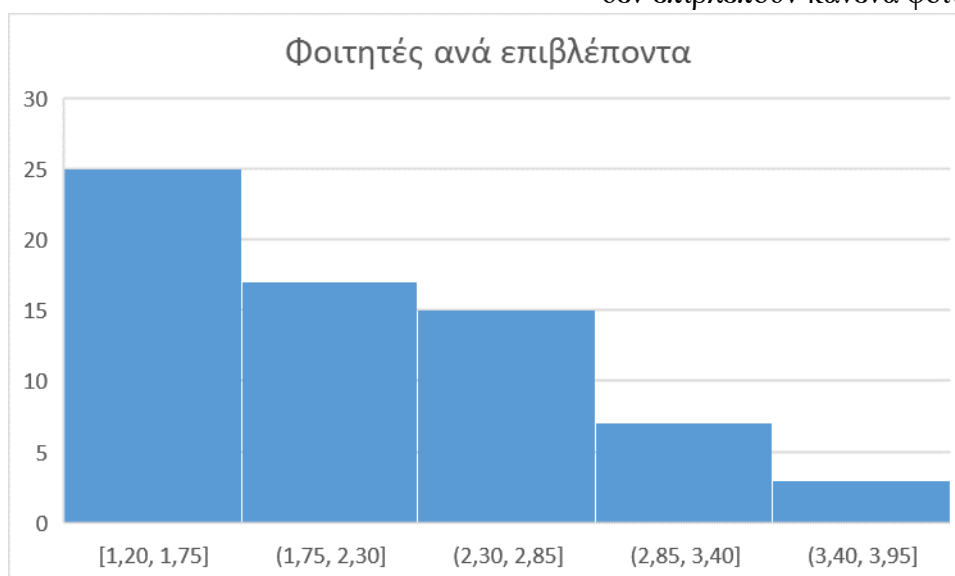
- **Φοιτητές χωρίς αντιπάλους:** Βρίσκει όλους τους υποψήφιους που δεν έχουν κανένα προτεινόμενο αντίπαλο.



Εικόνα 1

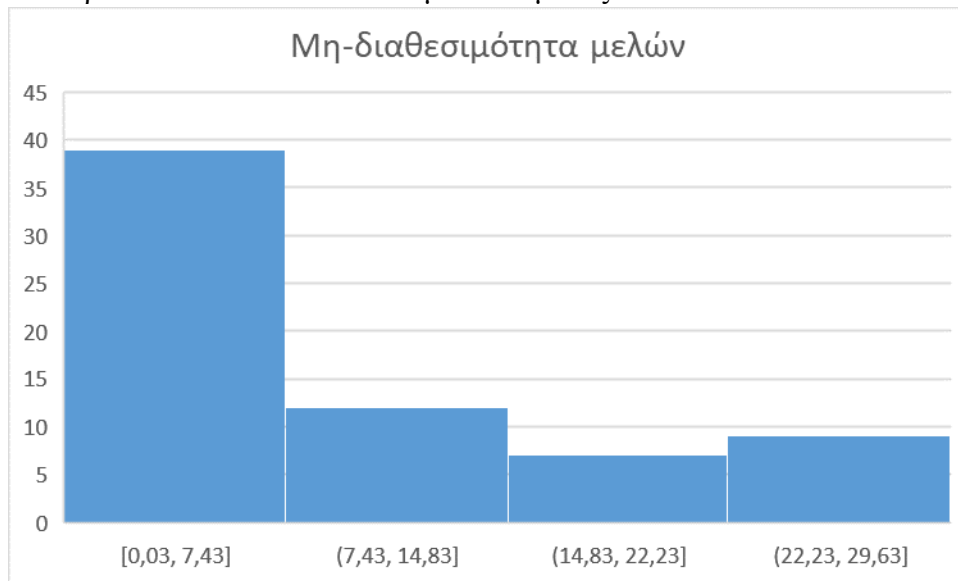
- **Φοιτητές ανά επιβλέποντα:** Βρίσκει το μέσον όρο των φοιτητών που αντιστοιχούν σε κάθε επιβλέποντα.

Φοιτ. ανά επιβλ. = $\frac{\text{πλήθος φοιτητών}}{(\text{πλήθος μελών} - \text{πλήθος μελών που δεν επιβλέπουν κανένα φοιτητή})}$



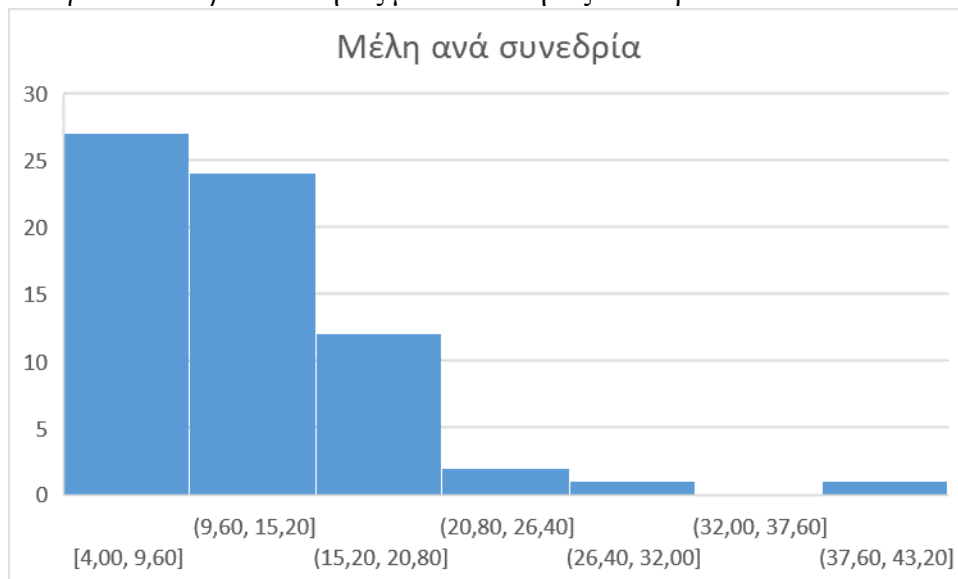
Εικόνα 2

- *Μη-διαθεσιμότητα μελών:* Υπολογίζει το μέσον όρο του πλήθους των συνεδριών που δεν είναι διαθέσιμο κάθε μέλος.



Εικόνα 3

- *Μέλη ανά συνεδρία =* πλήθος μελών / πλήθος συνεδριών



Εικόνα 4

- *Ποσοστό μελών χωρίς αντίπαλο* = φοιτητές χωρίς αντιπάλους / πλήθος φοιτητών



Εικόνα 5

- *Χωρητικότητα*: Βρίσκει το ποσοστό κάλυψης χώρου των συνεδριών.
 $\text{Χωρητ.} = \text{πλήθος φοιτητών} / \text{μέγιστο όριο φοιτητών} / \text{πλήθος συνεδριών}$



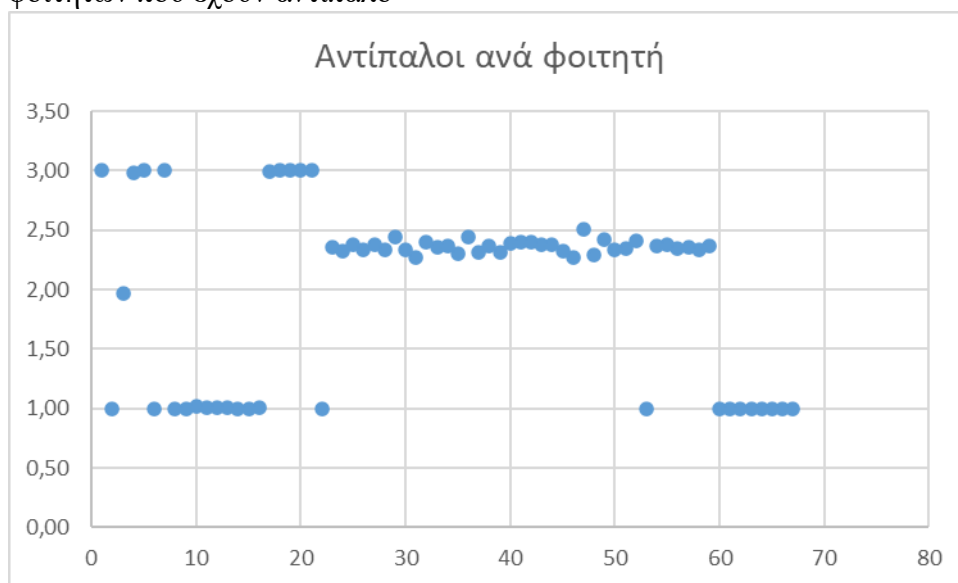
- *Ποσοστό μη-διαθεσιμότητας*: Υπολογίζει το ποσοστό του πλήθους των συνεδριών που δεν είναι διαθέσιμο κάθε μέλος.

Ποσοστό μη-διαθ. = Μη-διαθεσιμότητα μελών / πλήθος συνεδριών



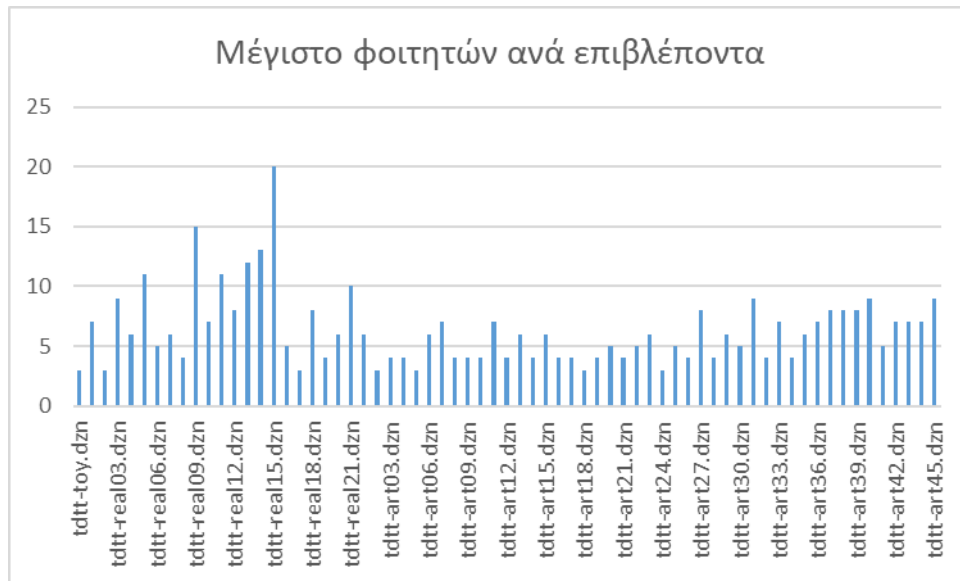
Εικόνα 6

- Αντίπαλοι ανά φοιτητή = άθροισμα των αντιπάλων κάθε φοιτητή / πλήθος φοιτητών που έχουν αντίπαλο



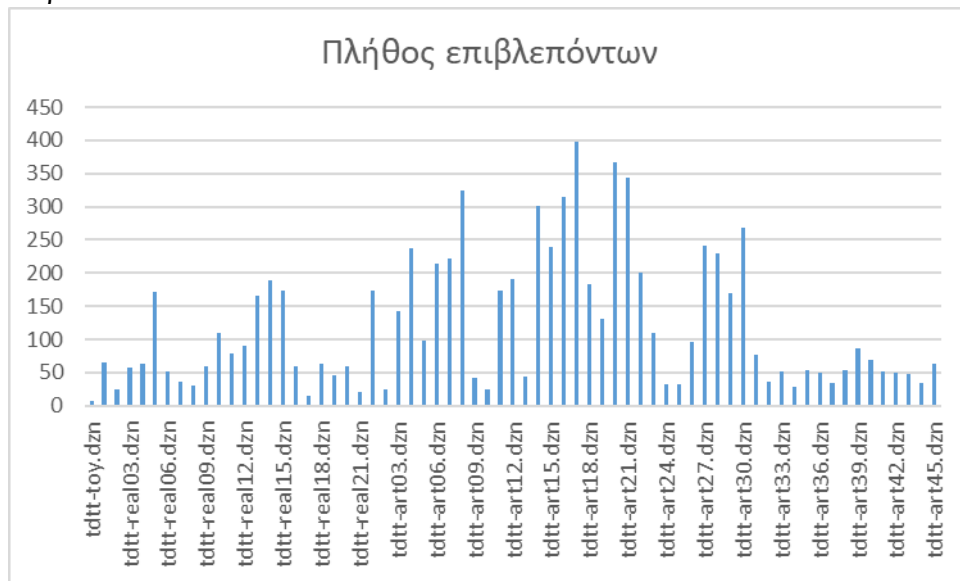
Εικόνα 7

- **Μέγιστο φοιτητών ανά επιβλέποντα:** Βρίσκει το μέγιστο αριθμό φοιτητών που μπορεί να επιβλέπει κάποιος επόπτης



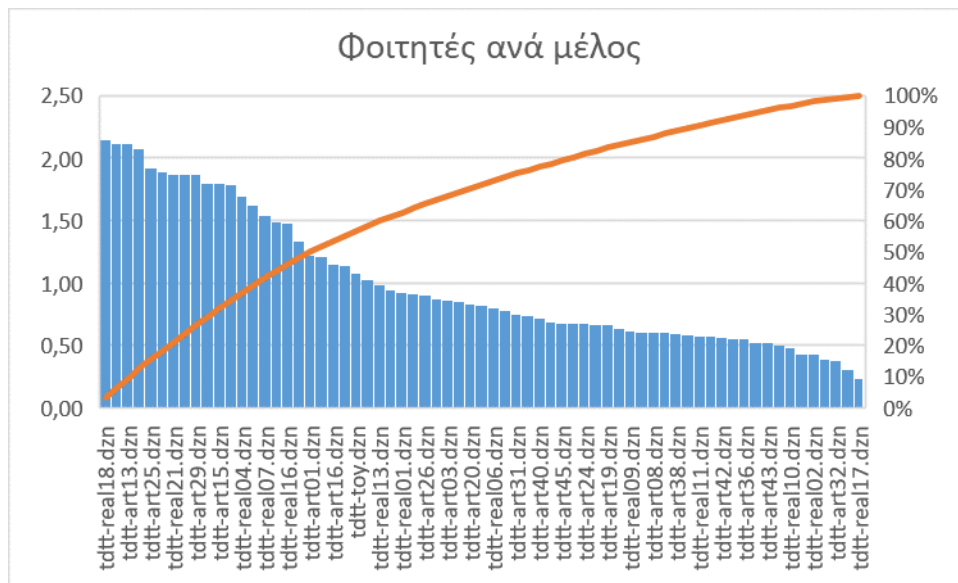
Εικόνα 8

- **Πλήθος επιβλεπόντων:** Μετράει πόσα από τα μέλη έχουν την αρμοδιότητα του επιβλέποντα.



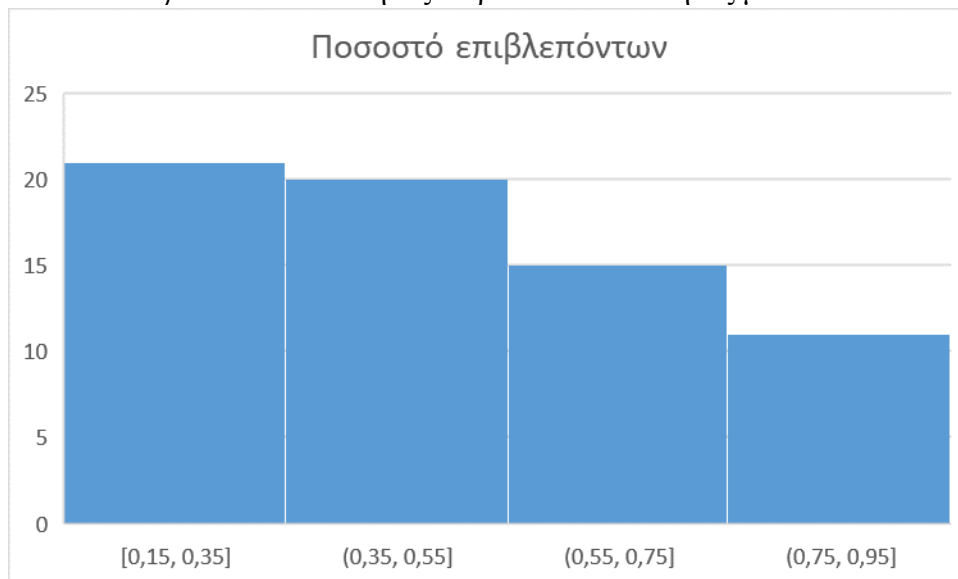
Εικόνα 9

- $\text{Φοιτητές ανά μέλος} = \text{πλήθος φοιτητών} / \text{πλήθος μελών}$



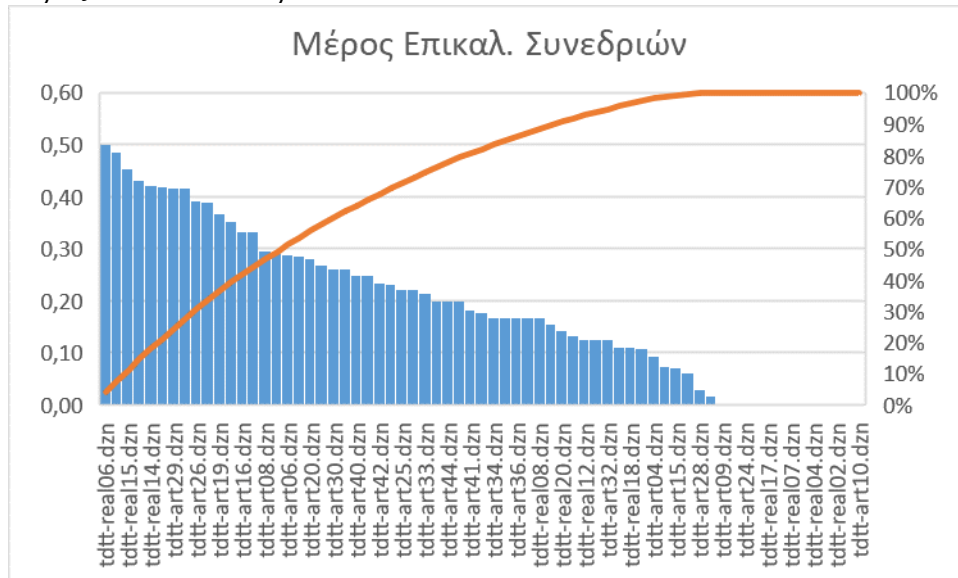
Εικόνα 10

- $\text{Ποσοστό επιβλεπόντων} = \text{πλήθος επιβλεπόντων} / \text{πλήθος μελών}$



Εικόνα 11

- *Μέρος Επικαλ. Συνεδριών:*



Εικόνα 12

Επιπλέον, παρουσιάζονται παρακάτω ο Πίνακας 2 και Πίνακας 3, όπου είναι συγκεντρωμένα όλα τα παραπάνω στατιστικών στοιχείων των δεδομένων με τις ακριβείς τιμές των επιπλέον χαρακτηριστικών που εξήγαμε για κάθε στιγμιότυπο.

Στιγμιότυπα	Φοιτητές χωρίς αντυπάλους	Φοιτητές ανά επιβλέποντα	Μη-διαθεσιμότητα μελών	Μέλη ανά συνεδρία	Ποσοστό μελών χωρίς αντίπαλο	Χωρητικότητα	Ποσοστό μη-διαθεσιμότητας
tdtt-toy.dzn	3	1,63	0,75	4,00	0,23	0,87	0,25
tdtt-real01.dzn	60	2,17	4,25	9,69	0,42	0,99	0,27
tdtt-real02.dzn	0	1,36	1,15	15,80	0,00	0,68	0,23
tdtt-real03.dzn	11	2,48	6,26	4,28	0,08	1,00	0,35
tdtt-real04.dzn	4	2,67	5,86	5,61	0,02	0,95	0,33
tdtt-real05.dzn	438	2,58	10,91	8,00	0,99	1,00	0,29
tdtt-real06.dzn	12	1,55	3,86	9,90	0,15	0,99	0,39
tdtt-real07.dzn	34	2,83	1,18	8,25	0,33	0,98	0,15
tdtt-real08.dzn	38	1,77	1,37	11,17	0,69	0,92	0,23
tdtt-real09.dzn	1	2,93	0,16	18,67	0,01	0,82	0,01
tdtt-real10.dzn	0	1,92	9,00	14,03	0,00	0,96	0,29
tdtt-real11.dzn	0	2,08	0,03	17,94	0,00	0,85	0,00
tdtt-real12.dzn	0	2,44	0,27	20,13	0,00	0,99	0,02
tdtt-real13.dzn	428	2,61	10,67	12,43	1,00	0,95	0,30
tdtt-real14.dzn	544	2,92	12,51	10,09	0,99	0,94	0,28
tdtt-real15.dzn	0	2,47	18,13	6,88	0,00	0,93	0,27
tdtt-real16.dzn	5	2,10	5,27	6,46	0,04	0,95	0,41
tdtt-real17.dzn	1	1,20	0,80	38,00	0,06	0,90	0,40
tdtt-real18.dzn	1	2,73	6,95	4,44	0,01	0,87	0,39
tdtt-real19.dzn	0	1,50	2,37	11,29	0,00	0,90	0,34
tdtt-real20.dzn	0	2,43	5,79	5,79	0,00	0,95	0,41
tdtt-real21.dzn	49	2,90	1,42	7,75	0,84	0,97	0,35
tdtt-art01.dzn	1	1,76	18,44	6,12	0,00	0,94	0,45
tdtt-art02.dzn	0	1,67	2,20	5,00	0,00	0,83	0,37
tdtt-art03.dzn	6	1,46	9,66	10,00	0,03	0,96	0,40
tdtt-art04.dzn	10	1,46	14,60	9,51	0,03	0,90	0,34
tdtt-art05.dzn	5	1,40	7,87	14,50	0,04	0,95	0,44
tdtt-art06.dzn	1	2,03	15,35	5,93	0,00	0,96	0,34
tdtt-art07.dzn	9	2,11	26,32	4,83	0,02	0,96	0,49
tdtt-art08.dzn	8	1,31	24,96	13,06	0,02	0,88	0,46
tdtt-art09.dzn	0	1,74	3,78	8,00	0,00	0,91	0,47
tdtt-art10.dzn	0	1,67	1,74	6,50	0,00	0,83	0,29
tdtt-art11.dzn	0	2,28	19,16	4,27	0,00	0,90	0,39
tdtt-art12.dzn	4	1,34	14,10	13,13	0,02	0,89	0,44
tdtt-art13.dzn	4	2,44	5,52	4,00	0,04	0,85	0,42
tdtt-art14.dzn	4	1,52	24,80	9,16	0,01	0,93	0,45
tdtt-art15.dzn	9	2,18	24,64	5,18	0,02	0,93	0,44
tdtt-art16.dzn	8	1,69	28,63	7,68	0,02	0,89	0,48
tdtt-art17.dzn	7	1,34	24,82	13,27	0,01	0,89	0,41
tdtt-art18.dzn	2	1,36	8,91	14,75	0,01	0,89	0,32
tdtt-art19.dzn	0	1,39	6,31	14,42	0,00	0,96	0,33
tdtt-art20.dzn	2	1,52	27,51	11,77	0,00	0,97	0,48
tdtt-art21.dzn	14	1,33	15,61	14,20	0,03	0,90	0,31
tdtt-art22.dzn	6	1,38	16,77	10,95	0,02	0,93	0,45
tdtt-art23.dzn	0	2,45	14,60	4,10	0,00	0,96	0,47
tdtt-art24.dzn	1	1,27	1,89	12,40	0,02	0,84	0,38
tdtt-art25.dzn	0	2,15	4,11	4,11	0,00	0,99	0,46
tdtt-art26.dzn	1	1,74	10,17	8,13	0,01	0,92	0,44
tdtt-art27.dzn	8	2,38	28,26	4,69	0,01	0,98	0,48
tdtt-art28.dzn	3	1,42	12,36	12,97	0,01	0,96	0,36
tdtt-art29.dzn	5	2,17	22,46	4,08	0,01	0,95	0,47
tdtt-art30.dzn	2	1,35	18,77	14,69	0,01	0,86	0,45
tdtt-art31.dzn	0	3,42	2,03	13,50	0,00	0,92	0,08
tdtt-art32.dzn	0	2,03	2,48	24,00	0,00	0,91	0,31
tdtt-art33.dzn	0	2,58	3,93	17,14	0,00	0,96	0,28
tdtt-art34.dzn	0	1,86	1,78	28,67	0,00	0,87	0,30
tdtt-art35.dzn	2	2,21	3,89	17,08	0,02	0,98	0,32
tdtt-art36.dzn	0	2,29	3,59	17,00	0,00	0,93	0,30
tdtt-art37.dzn	1	2,65	2,85	20,70	0,01	0,90	0,28
tdtt-art38.dzn	0	3,32	2,18	16,44	0,00	0,89	0,12
tdtt-art39.dzn	0	3,24	2,10	12,86	0,00	0,92	0,08
tdtt-art40.dzn	0	3,47	2,13	14,13	0,00	0,92	0,09
tdtt-art41.dzn	1	2,04	2,11	18,82	0,01	0,95	0,19
tdtt-art42.dzn	9	3,35	2,06	17,06	0,05	0,88	0,12
tdtt-art43.dzn	6	3,00	1,85	17,93	0,04	0,85	0,12
tdtt-art44.dzn	9	2,54	1,57	22,70	0,10	0,89	0,16
tdtt-art45.dzn	0	3,57	2,24	14,35	0,00	0,89	0,10

Πίνακας 2: Συγκεντρωτικός πίνακας στατιστικών στοιχείων των δεδομένων

Στιγμιότυπα	Αντίπαλοι ανά φοιτητή	Μέγιστο φοιτητών ανά επιβλέπον	Πλήθος επιβλεπόν των	Φοιτητές ανά μέλος	Ποσοστό επιβλεπόν των	Μέρος Επικαλ. Συνεδριών
tdtt-toy.dzn	3,00	3	8	1,08	0,67	0,33
tdtt-real01.dzn	1,00	7	66	0,92	0,43	0,00
tdtt-real02.dzn	1,97	3	25	0,43	0,32	0,00
tdtt-real03.dzn	2,98	9	58	1,87	0,75	0,00
tdtt-real04.dzn	3,00	6	64	1,69	0,63	0,00
tdtt-real05.dzn	1,00	11	171	1,49	0,58	0,00
tdtt-real06.dzn	3,00	5	51	0,80	0,52	0,50
tdtt-real07.dzn	1,00	6	36	1,55	0,55	0,00
tdtt-real08.dzn	1,00	4	31	0,82	0,46	0,17
tdtt-real09.dzn	1,02	15	59	0,62	0,21	0,13
tdtt-real10.dzn	1,00	7	109	0,48	0,25	0,42
tdtt-real11.dzn	1,01	11	79	0,57	0,28	0,13
tdtt-real12.dzn	1,00	8	91	0,69	0,28	0,13
tdtt-real13.dzn	1,00	12	165	0,99	0,38	0,49
tdtt-real14.dzn	1,00	13	189	1,21	0,42	0,42
tdtt-real15.dzn	1,01	20	173	0,94	0,38	0,45
tdtt-real16.dzn	2,99	5	59	1,48	0,70	0,00
tdtt-real17.dzn	3,00	3	15	0,24	0,20	0,00
tdtt-real18.dzn	3,00	8	63	2,15	0,79	0,11
tdtt-real19.dzn	3,00	4	46	0,87	0,58	0,00
tdtt-real20.dzn	3,00	6	60	1,80	0,74	0,14
tdtt-real21.dzn	1,00	10	20	1,87	0,65	0,25
tdtt-art01.dzn	2,35	6	174	1,22	0,69	0,07
tdtt-art02.dzn	2,33	3	24	1,33	0,80	0,17
tdtt-art03.dzn	2,38	4	142	0,87	0,59	0,42
tdtt-art04.dzn	2,33	4	238	0,85	0,58	0,09
tdtt-art05.dzn	2,38	3	98	0,52	0,38	0,11
tdtt-art06.dzn	2,34	6	214	1,63	0,80	0,29
tdtt-art07.dzn	2,44	7	221	1,79	0,85	0,35
tdtt-art08.dzn	2,34	4	325	0,61	0,46	0,30
tdtt-art09.dzn	2,27	4	42	1,14	0,66	0,00
tdtt-art10.dzn	2,40	4	24	1,03	0,62	0,00
tdtt-art11.dzn	2,36	7	173	1,89	0,83	0,06
tdtt-art12.dzn	2,36	4	191	0,61	0,45	0,16
tdtt-art13.dzn	2,30	6	45	2,12	0,87	0,23
tdtt-art14.dzn	2,44	4	301	0,91	0,60	0,11
tdtt-art15.dzn	2,31	6	240	1,80	0,83	0,07
tdtt-art16.dzn	2,37	4	314	1,15	0,68	0,33
tdtt-art17.dzn	2,32	4	397	0,67	0,50	0,02
tdtt-art18.dzn	2,39	3	183	0,60	0,44	0,18
tdtt-art19.dzn	2,40	4	132	0,67	0,48	0,37
tdtt-art20.dzn	2,39	5	366	0,83	0,55	0,28
tdtt-art21.dzn	2,38	4	344	0,63	0,48	0,00
tdtt-art22.dzn	2,38	5	200	0,68	0,49	0,43
tdtt-art23.dzn	2,32	6	110	2,12	0,87	0,29
tdtt-art24.dzn	2,27	3	33	0,68	0,53	0,00
tdtt-art25.dzn	2,51	5	33	1,92	0,89	0,22
tdtt-art26.dzn	2,29	4	97	0,90	0,52	0,39
tdtt-art27.dzn	2,42	8	242	2,08	0,87	0,39
tdtt-art28.dzn	2,34	4	230	0,74	0,52	0,03
tdtt-art29.dzn	2,34	6	169	1,87	0,86	0,42
tdtt-art30.dzn	2,41	5	268	0,59	0,43	0,26
tdtt-art31.dzn	1,00	9	77	0,75	0,22	0,27
tdtt-art32.dzn	2,37	4	36	0,38	0,19	0,13
tdtt-art33.dzn	2,37	7	52	0,56	0,22	0,21
tdtt-art34.dzn	2,35	4	28	0,30	0,16	0,17
tdtt-art35.dzn	2,36	6	53	0,57	0,26	0,17
tdtt-art36.dzn	2,33	7	49	0,55	0,24	0,17
tdtt-art37.dzn	2,37	8	34	0,43	0,16	0,20
tdtt-art38.dzn	1,00	8	53	0,59	0,18	0,22
tdtt-art39.dzn	1,00	8	87	0,78	0,24	0,29
tdtt-art40.dzn	1,00	9	70	0,72	0,21	0,25
tdtt-art41.dzn	1,00	5	51	0,50	0,25	0,18
tdtt-art42.dzn	1,00	7	49	0,57	0,17	0,24
tdtt-art43.dzn	1,00	7	47	0,52	0,17	0,20
tdtt-art44.dzn	1,00	7	35	0,39	0,15	0,20
tdtt-art45.dzn	1,00	9	63	0,68	0,19	0,26

Πίνακας 3: Συνέχεια Πίνακα 2

Κεφάλαιο 2

2.1 Χρονοπρογραμματισμός

Ο χρονοπρογραμματισμός είναι ένα από τα υπολογιστικά δύσκολα προβλήματα στον προγραμματισμό. Στον χρονοπρογραμματισμό, ο στόχος είναι να βρεθεί κατάλληλος χρόνος για μια σειρά εργασιών που απαιτούν περιορισμένους πόρους. Ανάλογα με τη φύση του προβλήματος, οι περιορισμοί μπορεί να διαφέρουν και να υπάρχουν πολλοί διαφορετικοί στόχοι. Για παράδειγμα, σε ορισμένες περιπτώσεις ο στόχος μπορεί να είναι η ελαχιστοποίηση της διάρκειας του συνολικού χρονικού διαστήματος κατά το οποίο πρόκειται να προγραμματιστούν οι εργασίες, σε άλλες περιπτώσεις ο στόχος μπορεί να είναι η εύρεση μιας εφικτής λύσης υπό την προϋπόθεση μιας καθορισμένης συνολικής χρονικής περιόδου και αρκετών άλλων περιορισμών. Ωστόσο, σε άλλες περιπτώσεις ο στόχος μπορεί να είναι η εύρεση λύσης κατά την οποία παραβιάζονται ελάχιστοι περιορισμοί.[2] Το χρονοδιάγραμμα λοιπόν είναι μια οργανωμένη λίστα, συνήθως παρουσιάζεται σε μορφή πίνακα, παρέχοντας πληροφορίες για μια σειρά οργανωμένων γεγονότων: ειδικότερα, την ώρα κατά την οποία προγραμματίζεται να πραγματοποιηθούν αυτά τα συμβάντα. Η γενική χρήση του χρονοπρογραμματισμού μπορεί να περιλαμβάνει τα ακόλουθα:

- 1) Σχολικό ωράριο, πίνακας συντονισμού μαθητών, καθηγητών, τάξεων και άλλων πόρων.
- 2) Πανεπιστημιακό ωράριο, πιο σύνθετο από το σχολικό και περιλαμβάνει το πρόγραμμα των διαλέξεων ή των εξετάσεων, ή και τον προγραμματισμό της παρουσίασης της διπλωματικής (όπως μελετάμε στο παρόν κείμενο).
- 3) Πρόγραμμα στο χώρο εργασίας, μια λίστα εργαζομένων που εργάζονται οποιαδήποτε δεδομένη ημέρα, εβδομάδα ή μήνα, π.χ. η ανάθεση βαρδιών σε νοσοκομεία.
- 4) Πίνακας δρομολογίων αεροπορικών εταιρειών, όπου σχεδιάζεται το πρόγραμμα των δρομολογίων καθορίζοντας τις πτήσεις που θα πραγματοποιηθούν σε μία συγκεκριμένη περίοδο.
- 5) Πίνακας δρομολογίων μέσων μαζικής μεταφοράς, κατάλογος των ωρών άφιξης και αναχώρησης των δημόσιων συγκοινωνιών σε συγκεκριμένες τοποθεσίες.
- 6) Πρόγραμμα αθλητικών εκδηλώσεων, μια λίστα με τις ώρες που θα παίξουν διαφορετικές αθλητικές ομάδες μεταξύ τους ως μέρος ενός πρωταθλήματος ή μιας σειράς αγώνων.

2.1.1 Προγραμματισμός εκπαιδευτικών ιδρυμάτων

Ο Προγραμματισμός εκπαιδευτικών ιδρυμάτων συνίσταται στην ανάθεση συναντήσεων δασκάλων/μαθητών σε ημέρες, ώρες και αίθουσες διδασκαλίας. Παρά τη φαινομενική αυτή απλότητα, η εμπειρία μας διδάσκει ότι κάθε ίδρυμα έχει τους δικούς του κανόνες, συμβάσεις και καθηλώσεις, καθιστώντας έτσι κάθε συγκεκριμένο πρόβλημα σχεδόν μοναδικό. Κατά συνέπεια, πολλές διαφορετικές διατυπώσεις προβλημάτων έχουν προταθεί στη βιβλιογραφία για το προγραμματισμό εκπαιδευτικών ιδρυμάτων, ανάλογα με τον τύπο του ιδρύματος (λύκειο, πανεπιστήμιο ή άλλο), το είδος των συναντήσεων (μαθήματα, διαλέξεις, εξετάσεις,...) και τις διαφορετικές ρυθμίσεις, περιορισμούς και στόχους[3].

2.1.2 Προβλήματα προγραμματισμού εκπαιδευτικών ιδρυμάτων

Σύμφωνα με τη βιβλιογραφία σχετικά με τον εκπαιδευτικό χρονοπρογραμματισμό, υπάρχουν τρία κύρια προβλήματα:

Χρονοπρογραμματισμός Λυκείου (HTT):

Ο εβδομαδιαίος προγραμματισμός για όλες τις τάξεις ενός Λυκείου, αποφεύγοντας οι καθηγητές να βρίσκονται σε δύο τάξεις ταυτόχρονα, και αντίστροφα.

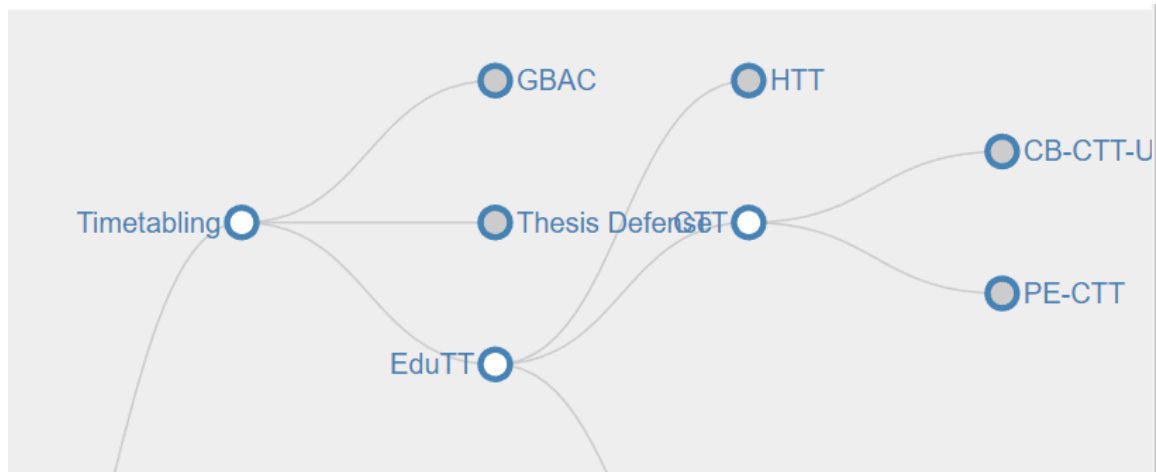
Χρονοπρογραμματισμός πανεπιστημιακών διαλέξεων (CTT):

Ο εβδομαδιαίος προγραμματισμός όλων των διαλέξεων ενός συνόλου πανεπιστημιακών μαθημάτων, αποφεύγοντας όσο το δυνατόν περισσότερο την επικάλυψη των διαλέξεων των μαθημάτων που έχουν κοινούς φοιτητές.

Χρονοπρογραμματισμός πανεπιστημιακών εξετάσεων (ETT):

Ο προγραμματισμός των εξετάσεων ενός συνόλου πανεπιστημιακών μαθημάτων, αποφεύγοντας την επικάλυψη των εξετάσεων των μαθημάτων που έχουν κοινούς φοιτητές και εκτείνοντας τις εξετάσεις για τους φοιτητές όσο το δυνατόν περισσότερο. Παρόλο που δεν είναι δυνατή η σαφής τομή μεταξύ HTT, CTT και ETT (π.χ., ορισμένα λύκεια οργανώνονται με πανεπιστημιακό τρόπο), συνήθως διαφέρουν σημαντικά μεταξύ τους και οι περισσότερες εργασίες στη βιβλιογραφία μπορούν να ταξινομηθούν σε ένα από αυτά τα τρία προβλήματα.

Εκτός από τα τρία κύρια προβλήματα στον τομέα του Χρονοπρογραμματισμού Εκπαιδευτικών Ιδρυμάτων υπάρχουν και άλλα παρόμοια προβλήματα στην βιβλιογραφία. Ανάμεσα σε αυτά είναι ο Χρονοπρογραμματισμός της Υπεράσπισης της Διατριβής (που εξετάζεται στην παρούσα εργασία), Διαχωρισμός Φοιτητών σε Τμήματα (Student Sectioning), Αναθέσεις Εκπαιδευτών/Ειδικευόμενων/Κατοίκων (για ιατρικές και στρατιωτικές σχολές) (Trainee/Intern/Resident Assignment), και Προγραμματισμός Συνεδρίων (Conference Scheduling)



Εικόνα 13

2.1.3 Σχετικές εργασίες

Στον ευρύ τομέα του εκπαιδευτικού χρονοπρογραμματισμού, όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, τρία προβλήματα που έχουν μελετηθεί περισσότερο είναι ο προγραμματισμός εξετάσεων και ο χρονοπρογραμματισμός μαθημάτων σε Γυμνάσιο/Λύκειο ή Πανεπιστήμιο. Ο ενδιαφερόμενος αναγνώστης μπορεί να πληροφορηθεί για λεπτομέρειες των διαφόρων παραλλαγών αυτών των προβλημάτων, των διαθέσιμων συνόλων δεδομένων και των προσεγγίσεων που έχουν χρησιμοποιηθεί για την αντιμετώπισή τους, συμβουλευόμενος τις διάφορες έρευνες που υπάρχουν, συμπεριλαμβανομένων των [4], [5], [3], [6] και [2].

Το πρόβλημα TDDT έχει παρουσιάσει λιγότερη προσοχή στη βιβλιογραφία. Αυτό μπορεί να συνέβη λόγω του μικρότερου αριθμού υποψηφίων που συνήθως επηρεάζονται από αυτό το είδος εξέτασης ανά πάσα στιγμή και των πολύ διαφορετικών κανόνων και κανονισμών που επιβάλλουν διάφορα ιδρύματα σχετικά με την εξέταση των διατριβών. Επιπλέον, ορισμένα πανεπιστήμια μπορεί να μην προκαθορίζουν μια συγκεκριμένη περίοδο του έτους για την υπεράσπιση της διατριβής, αλλά μπορεί να επιτρέψουν την παρουσίασή της καθ' όλη τη διάρκεια του ακαδημαϊκού έτους. Ωστόσο, καταφέραμε να εντοπίσουμε τα ακόλουθα έργα που σχετίζονται με το TDDT, πέρα από την κύρια πηγή μας [1]. Οι Bui και Hoang στο [7] πρότειναν μια πολυσκοπική παραλλαγή του προβλήματος υπεράσπισης της διατριβής. Η πολυαντικειμενική φύση του προβλήματος μελετήθηκε επίσης από τους Huynh et al. [8]. Μια εποικοδομητική ευρετική προσέγγιση προτάθηκε από τους Kochanikova και Rudova στο [9] που αναθέτει σε κάθε φοιτητή και επιτροπή μια συγκεκριμένη ημέρα και αίθουσα, ενώ προσπαθεί να επιτύχει σχεδόν ίσο φόρτο εργασίας για όλες τις επιτροπές. Μια άλλη παραλλαγή του προβλήματος από τον Dung et al. στο [10] στοχεύει στην ελαχιστοποίηση της διακύμανσης του φόρτου εργασίας των επιμέρους μελών της επιτροπής. Οι Limanto et al. στο [11] προτείνουν μια άλλη παραλλαγή του TDDT που αναθέτει μεμονωμένους ρόλους στα μέλη των επιτροπών και αναζητά λύσεις που

κατανέμουν εξίσου τον αριθμό των υποψηφίων που εξετάζει κάθε μέλος ΔΕΠ. Στο [12], οι Christopher και Wicaksana παρουσιάζουν μια άλλη παραλλαγή του TDDT και την λύνουν για μικρά μεγέθη προβλημάτων. Μια άλλη, πρόσφατη εργασία που μελετά μια πολυσκοπική παραλλαγή του προβλήματος είναι [13] από τους Almeida et al., που προσπαθεί να αποτυπώσει μια γενικότερη παραλλαγή του προβλήματος στοχεύοντας την εφαρμογή σε περισσότερες περιπτώσεις.

Δεδομένου ότι το TDDT στη βάση του είναι NP-Hard [1], δεν αποτελεί έκπληξη το γεγονός ότι έχουν δοκιμαστεί πολλές διαφορετικές προσεγγίσεις για την επίλυσή του. Έτσι, οι γενετικοί αλγόριθμοι, η βελτιστοποίηση σμήνους σωματιδίων, η τοπική αναζήτηση, ο μεικτός ακέραιος προγραμματισμός, ο προγραμματισμός με περιορισμούς και η μεταερευτική (π.χ. προσομοιωμένη ανόπτηση) φαίνεται να είναι οι πιο συνηθισμένοι τρόποι κατασκευής TDDT επιλυτών. Για τις πολυαντικειμενικές παραλλαγές του προβλήματος χρησιμοποιούνται γνωστοί πολυαντικειμενικοί αλγόριθμοι όπως ο Εξελικτικός Αλγόριθμος πολλαπλών στόχων (Direction-based Multi-objective Evolutionary Algorithm).

Ένα παρόμοιο πρόβλημα με το TDDT είναι το πρόβλημα προγραμματισμού συνεδριών [14]. Σε αυτό το πρόβλημα υπάρχουν δύο παραλλαγές, η παραλλαγή Presenter-Based Perspective (PBP) και η παραλλαγή Attender-Based Perspective (ABP). Στο PBP η διαθεσιμότητα των παρουσιαστών πρέπει να ικανοποιείται και οι επιπλέον προτιμήσεις χρόνου πρέπει να τηρούνται εάν είναι δυνατόν. Στο ABP, ο στόχος είναι η μεγιστοποίηση των παρευρισκομένων με βάση τις προτιμήσεις τους. Μια πρόσφατη εργασία που εξετάζει ένα συγκεκριμένο πρόβλημα προγραμματισμού συνεδριών χρησιμοποιώντας έναν αλγόριθμο που ενσωματώνει επαναλαμβανόμενη τοπική αναζήτηση, προσομοιωμένη ανόπτηση και δύο διαδικασίες που βασίζονται στον μαθηματικό προγραμματισμό είναι το [15].

2.2 Μαθηματικός Προγραμματισμός

Ο όρος «προγραμματισμός» χρησιμοποιήθηκε το 1940 για να περιγράψει το σχεδιασμό ένας προγράμματος δραστηριοτήτων σε ένα μεγάλο οργανισμό. Οι «προγραμματιστές» βρήκαν ότι μπορούσαν να αντιπροσωπεύουν το ποσό ή το επίπεδο κάθε δραστηριότητας ως μεταβλητή της οποίας η τιμή έπρεπε να καθοριστεί. Στη συνέχεια, θα μπορούσαν να περιγράψουν μαθηματικά τους περιορισμούς που είναι εγγενείς στο πρόβλημα προγραμματισμού ως ένα σύνολο εξισώσεων ή ανισώσεων που περιλαμβάνουν μεταβλητές. Μια λύση σε όλους αυτούς τους περιορισμούς θα θεωρηθεί ένα αποδεκτό χρονοδιάγραμμα. Σε γενικό πλαίσιο, η μοντελοποίηση ήταν αρκετά δύσκολη και πολύπλοκη διαδικασία απλά καθορίζοντας περιορισμούς. Η επιτυχία του προγραμματισμού εντέλει εξαρτιόταν από μια βασική γνώση που παρείχε έναν τρόπο να ξεπεραστεί αυτή η δυσκολία. Θα μπορούσε κανείς να καθορίσει, εκτός από κάποιους περιορισμούς, κι έναν στόχο: μια συνάρτηση μεταβλητών, το κόστος ή το κέρδος, που θα μπορούσε να χρησιμοποιηθεί για να αποφασιστεί εάν μια λύση ήταν καλύτερη από μια άλλη. Τότε δεν είχε σημασία ότι διαφορετικές λύσεις ικανοποιούσαν

τους περιορισμούς — ήταν αρκετό να βρεθεί μία τέτοια λύση που ελαχιστοποιούσε ή μεγιστοποιούσε τον στόχο. Ο όρος μαθηματικός προγραμματισμός άρχισε να χρησιμοποιείται για να περιγράψει την ελαχιστοποίηση ή τη μεγιστοποίηση μιας αντικειμενικής συνάρτησης πολλών μεταβλητών, και υπόκεινται σε περιορισμούς στις μεταβλητές.

Στην ανάπτυξη και εφαρμογή του μαθηματικού προγραμματισμού, ξεχωρίζει μια ειδική περίπτωση: αυτή στην οποία όλα τα κόστη, οι απαιτήσεις και άλλα ποσά ενδιαφέροντος είναι όροι αυστηρά ανάλογοι με τα επίπεδα των δραστηριοτήτων ή αθροίσματα τέτοιων όρων. Στη μαθηματική ορολογία, ο στόχος είναι μια γραμμική συνάρτηση και οι περιορισμοί είναι γραμμικές εξισώσεις και ανισώσεις. Ένα τέτοιο πρόβλημα ονομάζεται γραμμικό πρόγραμμα και η διαδικασία ρύθμισης ένας τέτοιου προβλήματος και επίλυσής του ονομάζεται *γραμμικός προγραμματισμός*. Ο γραμμικός προγραμματισμός είναι ιδιαίτερα σημαντικός επειδή μια μεγάλη ποικιλία προβλημάτων μπορούν να μοντελοποιηθούν ως γραμμικά προγράμματα και επειδή υπάρχουν γρήγορες και αξιόπιστες μέθοδοι για την επίλυση γραμμικών προγραμμάτων ακόμη και με χιλιάδες μεταβλητές και περιορισμούς. Οι ιδέες του γραμμικού προγραμματισμού είναι ένας σημαντικός για την ανάλυση και την επίλυση προβλημάτων μαθηματικού προγραμματισμού που δεν είναι γραμμικά. Η πρώτη επιτυχημένη υπολογιστική μέθοδος για γραμμικό προγραμματισμό, ο αλγόριθμος *simplex*, προτιμάτε αυτή τη στιγμή και αποτελεί αντικείμενο ολοένα και πιο αποτελεσματικών υλοποιήσεων. Παρά την ευρεία εφαρμογή του γραμμικού προγραμματισμού, η υπόθεση ένας γραμμικότητας είναι ένας φορές μη ρεαλιστική. Αν αντ' αυτού χρησιμοποιούνται κάποιες ομαλές μη γραμμικές συναρτήσεις των μεταβλητών στον στόχο ή ένας περιορισμός, το πρόβλημα ονομάζεται μη γραμμικό πρόγραμμα. Η επίλυση ενός τέτοιου προβλήματος είναι πιο δύσκολη, αν και στην πράξη δεν είναι και τόσο απίθανο να βρεθεί λύση. Αν και οι βέλτιστες τιμές των μη γραμμικών συναρτήσεων αποτελούν αντικείμενο μελέτης για πάνω από δύο αιώνες, οι υπολογιστικές μέθοδοι για την επίλυση μη γραμμικών προγραμμάτων σε ένας μεταβλητές αναπτύχθηκαν μόνο ένας τελευταίες δεκαετίες, μετά την επιτυχία των μεθόδων γραμμικού προγραμματισμού. Οι παραδοχές του γραμμικού προγραμματισμού καταρρέουν ένας εάν ορισμένες μεταβλητές πρέπει να λάβουν ακέραιες τιμές. Τότε το πρόβλημα ονομάζεται *ακέραιος προγραμματισμός* και γενικά γίνεται πολύ πιο δύσκολο. Ωστόσο, ένας συνδυασμός ταχύτερων υπολογιστών και πιο εξελιγμένων μεθόδων έχει κάνει τα μεγάλα προγράμματα ακέραιων αριθμών όλο και πιο προσβάσιμα τα τελευταία χρόνια.

2.2.1 Γραμμικός Προγραμματισμός

Το πιο γνωστό είδος μοντέλου βελτιστοποίησης είναι ο γραμμικός προγραμματισμός. Οι μεταβλητές ενός γραμμικού μοντέλου λαμβάνουν τιμές από κάποιο συνεχές εύρος. Ο στόχος και οι περιορισμοί πρέπει να χρησιμοποιούν μόνο γραμμικές συναρτήσεις των μεταβλητών. Τα γραμμικά μοντέλα είναι ιδιαίτερα σημαντικά γιατί αντιπροσωπεύουν με ακρίβεια πολλές πρακτικές εφαρμογές βελτιστοποίησης όπως εφαρμόζεται για την καθοδήγηση ποσοτικών αποφάσεων στον επιχειρηματικό

σχεδιασμό, στη βιομηχανική μηχανική και —σε μικρότερο βαθμό— στις κοινωνικές και φυσικές επιστήμες. Η απλότητα των γραμμικών συναρτήσεων καθιστά τα γραμμικά μοντέλα εύκολη στη διατύπωση, την ερμηνεία και την ανάλυση. Είναι επίσης εύκολο να λυθούν. Εάν μπορούμε να εκφράσουμε το πρόβλημά μας ως γραμμικό πρόγραμμα, ακόμη και σε χιλιάδες περιορισμούς και μεταβλητές, τότε μπορούμε να είμαστε σίγουροι ότι θα βρούμε τη βέλτιστη λύση με ακρίβεια και ταχύτητα.

Γραμμικός προγραμματισμός, λοιπόν είναι η τεχνική μαθηματικής μοντελοποίησης στην οποία μια γραμμική συνάρτηση μεγιστοποιείται ή ελαχιστοποιείται όταν υπόκειται σε διάφορους περιορισμούς. Η λύση ενός προβλήματος γραμμικού προγραμματισμού περιορίζεται στην εύρεση της βέλτιστης τιμής (μεγαλύτερης ή μικρότερης, ανάλογα με το πρόβλημα) της γραμμικής έκφρασης (που ονομάζεται αντικειμενική συνάρτηση):

$$f = c_1x_1 + \dots + c_nx_n$$

υπόκεινται σε ένα σύνολο περιορισμών που εκφράζονται ως ανισότητες:

$$\begin{aligned} a_{11}x_1 + \dots + a_{1n}x_n &\leq b_1 \\ &\vdots \\ a_{m1}x_1 + \dots + a_{mn}x_n &\leq b_m \text{ with } \forall x_i \geq 0. \end{aligned}$$

Τα a , b και c είναι σταθερές που καθορίζονται από τις δυνατότητες, τις ανάγκες, το κόστος, τα κέρδη και άλλες απαιτήσεις και περιορισμούς του προβλήματος. Η βασική υπόθεση στην εφαρμογή αυτής της μεθόδου είναι ότι οι διάφορες σχέσεις μεταξύ ζήτησης και διαθεσιμότητας είναι γραμμικές. Δηλαδή, κανένα από τα x_i δεν ανυψώνεται σε δύναμη διαφορετική από το 1. Για να ληφθεί η λύση σε αυτό το πρόβλημα, είναι απαραίτητο να βρεθεί η λύση του συστήματος των γραμμικών ανισοτήτων (δηλαδή, το σύνολο των n τιμών των μεταβλητών x_i που ικανοποιεί ταυτόχρονα όλες τις ανισότητες). Στη συνέχεια, η αντικειμενική συνάρτηση αξιολογείται αντικαθιστώντας τις τιμές του x_i στην εξίσωση που ορίζει τη f .

2.2.2 Ακέραιος Προγραμματισμός

Τα μοντέλα γραμμικού προγραμματισμού είναι συνεχή, με την έννοια ότι οι μεταβλητές απόφασης επιτρέπεται να είναι κλασματικές. Συχνά αυτό δεν είναι μια ρεαλιστική υπόθεση, διότι υπάρχουν περιπτώσεις όπου κάποιες μεταβλητές πρέπει υποχρεωτικά να παίρνουν ακέραιες τιμές. Τότε το πρόβλημα λέγεται πρόβλημα Ακέραιου Προγραμματισμού (IP). Τόσο η αντικειμενική συνάρτηση όσο και οι περιορισμοί πρέπει να είναι γραμμικοί. Η πιο συχνά χρησιμοποιούμενη μέθοδος για την επίλυση ενός προβλήματος ακέραιου προγραμματισμού είναι η μέθοδος διακλάδωσης και δέσμευσης (branch-and-bound). Η επίλυση τέτοιου είδους προβλημάτων χρειάζονται γενικά περισσότερο χρόνο σε σχέση με την επίλυση προβλημάτων γραμμικού προγραμματισμού ίδιου μεγέθους.

2.2.3 Μικτός Ακέραιος Προγραμματισμός

Ένα μικτό γραμμικό πρόβλημα είναι ένα πρόβλημα βελτιστοποίησης στο οποίο υπάρχουν ένα μη κενό υποσύνολο ακέραιων μεταβλητών (άγνωστες) και ένα υποσύνολο πραγματικών (συνεχών) μεταβλητών, οι περιορισμοί είναι όλοι γραμμικές εξισώσεις ή ανισώσεις, και η αντικειμενική συνάρτηση είναι γραμμική με σκοπό την ελαχιστοποίηση (ή τη μεγιστοποίησή) της. Στη συγκεκριμένη μελέτη, λόγω του ότι έχουμε τετραγωνική αντικειμενική συνάρτηση, το πρόβλημα βελτιστοποίησης είναι Μικτού ακέραιου τετραγωνικού προγραμματισμού (MIQP).[16]

CPLEX

Το IBM ILOG CPLEX προσφέρει βιβλιοθήκες C, C++, Java, .NET και Python που επιλύουν γραμμικό προγραμματισμό (LP) και συναφή προβλήματα. Συγκεκριμένα, επιλύει προβλήματα βελτιστοποίησης γραμμικά ή τετραγωνικά με περιορισμούς, όπου ο στόχος που πρόκειται να βελτιστοποιηθεί μπορεί να εκφραστεί ως γραμμική ή κυρτή τετραγωνική συνάρτηση. Οι μεταβλητές στο μοντέλο μπορούν να δηλωθούν ως συνεχείς ή περαιτέρω περιορισμένες ώστε να λαμβάνουν μόνο ακέραιες τιμές.

Περιγραφή επίλυσης προβλημάτων τετραγωνικού προγραμματισμού (QPs) με CPLEX

Η CPLEX λύνει τετραγωνικά προγράμματα, δηλαδή ένα μοντέλο στο οποίο οι περιορισμοί είναι γραμμικοί, αλλά η αντικειμενική συνάρτηση μπορεί να περιέχει έναν ή περισσότερους τετραγωνικούς όρους. Αυτά τα προβλήματα είναι επίσης γνωστά ως QP. Όταν τέτοια προβλήματα είναι κυρτά, η CPLEX τα λύνει συνήθως αποτελεσματικά σε πολωνυμικό χρόνο. Τα μη κυρτά QP, ωστόσο, είναι γνωστό ότι είναι αρκετά δύσκολα. Θεωρητικά χαρακτηρίζονται ως NP-hard. Η CPLEX εφαρμόζει διάφορες προσεγγίσεις σε αυτά τα προβλήματα, όπως με προσέγγιση φραγμένων αλγορίθμων ή branch and bound αλγορίθμων. Σημειωτέων, στην προσέγγιση branch and bound, δεν υπάρχει θεωρητική εγγύηση για την πολυπλοκότητα ενός τέτοιου προβλήματος. Κατά συνέπεια, η λύση ενός τέτοιου προβλήματος (δηλαδή ενός μη κυρτού QP) μπορεί να πάρει πολλές τάξεις μεγέθους περισσότερο από τη λύση ενός κυρτού QP συγκρίσιμων διαστάσεων.

Η ταξινομημένη λίστα κωδικών της κατάστασης (status) των λύσεων που παρέχεται από την CPLEX εμφανίζεται στον παρακάτω πίνακα:

1	Βρέθηκε η βέλτιστη λύση
2	Μη φραγμένο πρόβλημα
3	Μη εφικτή λύση
4	Πρόβλημα ανέφικτο ή μη φραγμένο

Πίνακας 4: Ταξινομημένη λίστα κωδικών της κατάστασης που παρέχεται από τη CPLEX

2.3 Προγραμματισμός υπό περιορισμούς

Ο Προγραμματισμός υπό περιορισμούς (Constraint Programming – εν συντομία CP) έχει αναπτυχθεί από την Τεχνική Νοημοσύνη και την Επιστήμη των Υπολογιστών για την επίλυση προβλημάτων ικανοποίησης περιορισμών (Constraint Satisfaction Problems - CSP). Ο CP έχει βρει εφαρμογές σε ποικίλους τομείς και ειδικότερα έχει σημειώσει μεγάλη επιτυχία σε τομείς που περιλαμβάνουν συνδυαστική βελτιστοποίηση, όπως προβλήματα ανάθεσης και προγραμματισμού. Η λύση ενός προβλήματος βελτιστοποίησης από τον CP ξεκινά με τη μετατροπή του προβλήματος βελτιστοποίησης σε ένα κατάλληλο CSP που αντιμετωπίζει την αντικειμενική συνάρτηση ως περιορισμό και αξιολογεί την τιμή της αντικειμενικής συνάρτησης σε κάθε εφικτή λύση. Αυτό ακολουθείται από την απόρριψη των ανέφικτων τιμών από τον χώρο των μεταβλητών με τη διάδοση του συνόλου των περιορισμών.

Δύο από τα πιο σημαντικά χαρακτηριστικά του είναι η εκφραστική του δύναμη μοντελοποίησης και η ικανότητα να εγγυάται τον προσδιορισμό βέλτιστων λύσεων ακόμη και για μη-κυρτά και μη-γραμμικά ακέραια προβλήματα προγραμματισμού. Η εκφραστική δύναμη μοντελοποίησης επιτρέπει την ανάπτυξη συμπαγών και "φυσικών" μοντέλων ενώ η μείωση του χώρου μέσω της διάδοσης των περιορισμών εξασφαλίζει τον προσδιορισμό της παγκοσμίως βέλτιστης λύσης χωρίς να παγιδευεται σε τοπικές βέλτιστες λύσεις. Πρόσθετα χαρακτηριστικά όπως ο αποτελεσματικός προσδιορισμός των πολλαπλών λύσεων και οι σχεδόν καλύτερες λύσεις καθιστούν την μια ελκυστική τεχνική βελτιστοποίησης. Ωστόσο, η CP απαιτεί ένα ρητό πλαίσιο βελτιστοποίησης για τη διάδοση περιορισμών για τη μείωση του χώρου των μεταβλητών, μπορεί επίσης περιστασιακά να απαιτήσει σημαντική ποσότητα υπολογιστικών πόρων για την επιβεβαίωση της βέλτιστης λύσης, καθώς δεν βασίζεται σε όρια/χαλάρωση ή παράγωγες πληροφορίες.

Η Google παρέχει κάποιους τρόπους επίλυσης προβλημάτων CP:

- CP-SAT επιλυτής
- Πρωτότυπος CP επιλυτής

Δύο κλασικά προβλήματα CP είναι το πρόβλημα N-queens και τα κρυπτοαριθμητικά παζλ.

Στην παρούσα εργασία γίνεται χρήση του CP-SAT επιλυτή για την εύρεση λύσης για το πρόβλημα υπεράσπισης της διατριβής. Το πρόγραμμα επίλυσης CP-SAT επιστρέφει μία από τις τιμές κατάστασης που εμφανίζονται στον παρακάτω πίνακα.

OPTIMAL	Μία βέλτιστη εφικτή λύση βρέθηκε.
FEASIBLE	Μία εφικτή λύση βρέθηκε, αλλά δεν ξέρουμε αν είναι βέλτιστη.
INFEASIBLE	Το πρόβλημα αποδείχθηκε ανέφικτο.
MODEL_INVALID	Μη έγκυρο πρόβλημα.
UNKNOWN	Άγνωστο, διότι καμιά λύση δεν βρέθηκε, πριν κάτι να προκαλέσει στο λύτη να σταματήσει.

Πίνακας 5: Τιμές κατάστασης που εμφανίζει το πρόγραμμα επίλυσης CP-SAT

2.4 Ευρετικοί Αλγόριθμοι

Η ευρετική τεχνική δεν είναι κάτι καινούργιο, παίζει σημαντικό ρόλο στην καθημερινή μας ζωή, τόσο στην επίλυση προβλημάτων όσο και στη λήψη αποφάσεων βάσει δεδομένων. Καθώς στις μέρες μας, ο κόσμος είναι γεμάτος πληροφορίες και ο εγκέφαλός μας είναι ικανός να επεξεργαστεί μόνο ένα ορισμένο ποσό από αυτές, η ευρετική μας βοηθάει πολύ. Γιατί αν προσπαθούσαμε να αναλύσουμε κάθε πτυχή από κάθε κατάσταση ή απόφαση, δεν θα καταφέρναμε ποτέ τίποτα. Λαμβάνουμε χιλιάδες αποφάσεις κάθε μέρα και τις περισσότερες από αυτές δεν τις σκεφτόμαστε πραγματικά, «ξέρουμε» πώς να συμπεριφερόμαστε σε συγκεκριμένες καταστάσεις με βάση την εμπειρία μας και αυτό είναι το θέμα της ευρετικής. Όταν προσπαθούμε να λύσουμε ένα πρόβλημα ή να πάρουμε μια απόφαση, συχνά στρεφόμαστε σε αυτές τις τεχνικές συντομότερου δρόμου όταν χρειαζόμαστε μια γρήγορη λύση.

Όσον αφορά τον ευρετικό προγραμματισμό, μας βοηθά να λαμβάνουμε αποφάσεις και κρίσεις γρήγορα χωρίς να χρειάζεται να ξοδεύουμε πολύ χρόνο στην έρευνα και την ανάλυση πληροφοριών. Συνήθως περιλαμβάνουν την εστίαση σε μια πτυχή ενός σύνθετου προβλήματος και την παράβλεψη άλλων. Λειτουργούν καλά στις περισσότερες περιπτώσεις, αλλά μπορεί να οδηγήσουν σε συστηματικές αποκλίσεις από τη λογική, την πιθανότητα ή την ορθολογική επιλογή. Παραδείγματα που χρησιμοποιούν ευρετικές μεθόδους περιλαμβάνουν τη χρήση ενός εμπειρικού κανόνα, μιας μελετημένης εικασίας, μιας διαισθητικής κρίσης, δημιουργίας στερεοτύπων, δημιουργίας προφίλ ή της κοινής λογικής.

Στην επιστήμη των υπολογιστών, την τεχνητή νοημοσύνη και τη μαθηματική βελτιστοποίηση, η ευρετική είναι μια τεχνική που έχει σχεδιαστεί για την ταχύτερη επίλυση ενός προβλήματος όταν οι κλασικές μέθοδοι είναι πολύ αργές ή για την εύρεση μιας κατά προσέγγιση λύσης όταν οι κλασικές μέθοδοι αποτυγχάνουν να βρουν κάποια ακριβή λύση. Αυτό επιτυγχάνεται με τη βέλτιστη ανταπόκριση, την πληρότητα, την ακρίβεια ή την συνέπεια ως προς την ταχύτητα.

Ο ευρετικός προγραμματισμός χρησιμοποιεί μια πρακτική μέθοδο, που δεν εγγυάται ότι είναι βέλτιστη, τέλεια, λογική ή ορθολογική, αλλά αντ' αυτού επαρκής για την επίτευξη ενός άμεσου στόχου. Είναι σημαντικό να τονιστεί ότι οι ευρετικές είναι οι στρατηγικές που προέρχονται από προηγούμενες εμπειρίες σε παρόμοια προβλήματα. Αυτές οι στρατηγικές βασίζονται στη χρήση εύκολα προσβάσιμων, αν και ελάχιστα εφαρμόσιμων, πληροφοριών για τον έλεγχο της επίλυσης προβλημάτων σε ανθρώπους, μηχανές και αφηρημένα ζητήματα. Και ο στόχος μιας ευρετικής είναι να παράγει μια λύση σε εύλογο χρονικό πλαίσιο που είναι αρκετά καλή για την επίλυση του προβλήματος.

Ειδικότερα, ένας ευρετικός αλγόριθμος, σε αντίθεση με τον αλγοριθμικό προγραμματισμό ο οποίος βασίζεται σε μαθηματικά αποδείξιμες διαδικασίες, είναι αυτός που έχει σχεδιαστεί για να λύνει ένα πρόβλημα με πιο γρήγορο και αποτελεσματικό τρόπο από τις παραδοσιακές μεθόδους. Οι ευρετικοί αλγόριθμοι χρησιμοποιούνται συχνά για την επίλυση προβλημάτων NP-complete, μια κατηγορία προβλημάτων απόφασης, όπου δεν υπάρχει γνωστός τρόπος για να βρεθεί μια λύση γρήγορα και με ακρίβεια, αν και οι λύσεις μπορούν να επαληθευτούν όταν δίνονται. Οι ευρετικές μπορούν να παράγουν μια λύση μεμονωμένα ή να χρησιμοποιηθούν για να παρέχουν μια καλή βάση και συμπληρώνονται με αλγόριθμους βελτιστοποίησης.

Αυτοί οι αλγόριθμοι χρησιμοποιούνται συχνότερα όταν οι κατά προσέγγιση λύσεις είναι επαρκείς και οι ακριβείς λύσεις είναι υπολογιστικά πολύ ακριβές. Κάποιοι γνωστοί ευρετικοί αλγόριθμοι που χρησιμοποιούνται συνήθως είναι οι:

- Αλγόριθμοι Νοημοσύνης σμήνους
- Αλγόριθμοι Αναζήτησης Tabu
- Αλγόριθμοι Προσομοιωμένης ανόπτησης
- Γενετικοί αλγόριθμοι
- Μηχανές Διανυσμάτων Υποστήριξης

Συμπερασματικά, οι ευρετικοί είναι οι αλγόριθμοι, που χρησιμοποιούν εμπειρικούς κανόνες για την επίλυση προβλημάτων, αναζητώντας ποιοτικές εφικτές λύσεις που ικανοποιούν τους περιορισμούς του προβλήματος. Συνήθως υστερούν στην ποιότητα της λύσης, αλλά υπερέχουν στον απαιτούμενο χρόνο υπολογισμού. Έτσι η ευρετική τεχνική είναι ευρέως γνωστή για τη χρήση της για ποικίλους λόγους.[17], [18]

2.5 Μεταευρετικοί Αλγόριθμοι

Οι ευρετικές, που αναφερθήκαμε στο προηγούμενο κεφάλαιο, και οι μεταευρετικές τεχνικές βελτιστοποίησης αναπτύχθηκαν στα τέλη της δεκαετίας του '70 για να ξεπεραστούν τα μειονεκτήματα των LP, NLP και DP. Η ευρετική είναι μια τεχνική που στοχεύει στην ταχύτερη επίλυση ενός προβλήματος όταν οι παραδοσιακές τεχνικές είναι πολύ αργές. Μια μεταευρετική είναι μια τεχνική υψηλότερου επιπέδου που αναζητά, δημιουργεί ή επιλέγει μια ευρετική που μπορεί να παρέχει μια αρκετά καλή λύση σε ένα πρόβλημα βελτιστοποίησης. Αυτές οι νεότερες τεχνικές αναπτύχθηκαν για την αντιμετώπιση προβλημάτων όπως η μη γραμμικότητα, οι πολλαπλοί στόχοι, η αβεβαιότητα και ούτω καθεξής. Γενικά, ταξινομούνται σε αλγόριθμους βασισμένους στον πληθυσμό και σε αλγόριθμους με βάση τη γειτονιά. Οι αλγόριθμοι που βασίζονται στον πληθυσμό είναι ο εξελικτικός υπολογισμός και η νοημοσύνη σμήνους. Αυτές οι τεχνικές είναι ευέλικτες και μπορούν να προσφέρουν μια καθολική λύση με μεγαλύτερη ευκολία. Τα αποτελέσματα εξαρτώνται από τον αρχικό πληθυσμό, ο οποίος δημιουργείται τυχαία για να ληφθεί η συνολική λύση που βασίζεται στη θεωρία πιθανοτήτων. Οι εξελικτικοί υπολογισμοί περιλαμβάνουν γενετικό αλγόριθμο (GA), διαφορική εξέλιξη (DE), γενετικό προγραμματισμό (GP), εξελικτικό προγραμματισμό (EP) και εξελικτικές στρατηγικές (ES). Οι αλγόριθμοι βασισμένοι στη νοημοσύνη σμήνους περιλαμβάνουν βελτιστοποίηση αποικίας μυρμηγκιών (ACO), αναζήτηση αρμονίας (HS), βελτιστοποίηση σμήνους σωματιδίων (PSO), αναζήτηση κούκου (CS), τεχνητή αποικία μελισσών (ABC), αλγόριθμος πυγολαμπίδας (FA), αλγόριθμος νυχτερίδων (BA), βελτιστοποίηση ζευγαρώματος μέλισσας (HBMO) και αλγόριθμος ανακατεμένου άλματος βατράχου (SFLA). Οι αλγόριθμοι που βασίζονται σε γειτονιά περιλαμβάνουν προσομοιωμένη ανόπτηση (SA) και αναζήτηση tabu (TS).

Ένας μεταευρετικός αλγόριθμος είναι μια διαδικασία αναζήτησης που έχει σχεδιαστεί για την εύρεση μιας καλής λύσης σε ένα πρόβλημα βελτιστοποίησης το οποίο είναι πολύπλοκο και δύσκολο να λυθεί ως προς τη βέλτιστη λύση. Είναι επιτακτική ανάγκη

να βρεθεί μια σχεδόν βέλτιστη λύση που βασίζεται σε ατελείς ή ελλιπείς πληροφορίες του πραγματικού κόσμου των περιορισμένων πόρων (π.χ. υπολογιστικής ισχύς και χρόνος). Η εμφάνιση της μεταερευνητικής για την επίλυση τέτοιων προβλημάτων βελτιστοποίησης είναι ένα από τα πιο αξιοσημείωτα επιτεύγματα των δύο τελευταίων δεκαετιών στην επιχειρησιακή έρευνα.

Δημοφιλείς μεταερευνητικές μέθοδοι για συνδυαστικά προβλήματα περιλαμβάνουν προσομοιωμένη ανόπτηση και αναρρίχηση λόφων (hill climbing) από τους Kirkpatrick et al., γενετικούς αλγόριθμους από Holland et al., αναζήτηση διασποράς και αναζήτηση tabu από Glover, κ.α.[17]

2.6 Προσομοιωμένη Ανόπτηση

Η προσομοιωμένη ανόπτηση είναι μια μέθοδος για την ελαχιστοποίηση συναρτήσεων ενός μεγάλου πλήθους μεταβλητών και άλλων θεμάτων συνδυαστικής βελτιστοποίησης. Λόγω του γεγονότος ότι πολλά ζητήματα σχεδιασμού πραγματικού κόσμου μπορεί να αντιπροσωπεύονται ως τέτοια ζητήματα βελτιστοποίησης, υπάρχει μεγάλο ενδιαφέρον για ευρείες μεθόδους επίλυσής τους. Μια τέτοια σχετικά νέα τεχνική είναι η προσομοιωμένη ανόπτηση. Είναι εμπνευσμένη από τη σύγκριση με την ανόπτηση των στερεών με στατιστική μηχανική. Αυτό το ζήτημα φυσικής είναι ενδιαφέρον διότι: τα μέταλλα έχουν δύο χαρακτηριστικά, σκληρότητα και ευθραυστότητα, ανταγωνιστικά μεταξύ τους. Ανόπτηση είναι η διαδικασία θέρμανσης του μετάλλου και έπειτα η σταδιακή ψύξη του με ελεγχόμενο ρυθμό ψύξης, με σκοπό το μέταλλο να κατέχει την ιδανική αναλογία των χαρακτηριστικών της σκληρότητας και ευθραυστότητας. Η Ανόπτηση επηρεάζει την κρυσταλλική δομή του μετάλλου και προσδίδει στο μέταλλο τις επιθυμητές ιδιότητες. Αν η ψύξη γίνει με πιο γρήγορο ή πιο αργό ρυθμό από ότι είναι αναγκαίο, δεν επιτυγχάνεται το επιθυμητό αποτέλεσμα.

Ισοδύναμα με την διαδικασία της ανόπτησης στην κατεργασία του μετάλλου (από όπου και το όνομα της μεθόδου) είναι η προσομοιωμένη ανόπτηση ως μια διαδικασία δημιουργίας ευριστικών λύσεων που βασίζεται σε λογική και κανόνες για να αλλάξει επανειλημμένα μια υποβέλτιστη λύση σε ένα πρόβλημα και επιδιώκει να εντοπίσει την καλύτερη δυνατή λύση, συνήθως μία κοντινή λύση στη βέλτιστη. Η διαδικασία είναι σχετικά γρήγορη, σε σύγκριση με τις παραδοσιακές μαθηματικές μεθόδους προγραμματισμού καθώς και με άλλες ευρετικές διαδικασίες. Οι διορθώσεις σε μια λύση επιλέγονται τυχαία από τη γειτονιά της υπάρχουσας λύσης, και αν έχουν ως αποτέλεσμα μια κατώτερη λύση, γίνονται αποδεκτές στην αρχή, αλλά η πιθανότητα αποδοχής μειώνεται καθώς ο αριθμός των προσαρμογών αυξάνεται.

Το ενδιαφέρον για τέτοιες τεχνικές λύσης είναι έντονο επειδή λίγα σημαντικά προβλήματα συνδυαστικής βελτιστοποίησης μπορούν να επιλυθούν ακριβώς σε εύλογο χρόνο. Πολλά προβλήματα βελτιστοποίησης που προκύπτουν στην πράξη είναι NP-complete, όλες οι γνωστές τεχνικές για την επίτευξη μιας ακριβούς λύσης απαιτούν έναν εκθετικά αυξανόμενο αριθμό βημάτων καθώς τα προβλήματα γίνονται μεγαλύτερα. Ως εκ τούτου, έχει δοθεί έμφαση στις ευρετικές τεχνικές επίλυσης αυτών των σημαντικών προβλημάτων, όπως το πρόβλημα του πλανόδιου πωλητή.

Το βασικό πλεονέκτημα του αλγορίθμου είναι η ικανότητά του να μην εγκλωβίζεται σε τοπικά βέλτιστα. Αυτό γίνεται εφικτό μέσω της αποδοχής με κάποια πιθανότητα και λύσεων που δεν βελτιώνουν την αντικειμενική συνάρτηση.[19], [20]

2.7 Γενετικός Αλγόριθμος

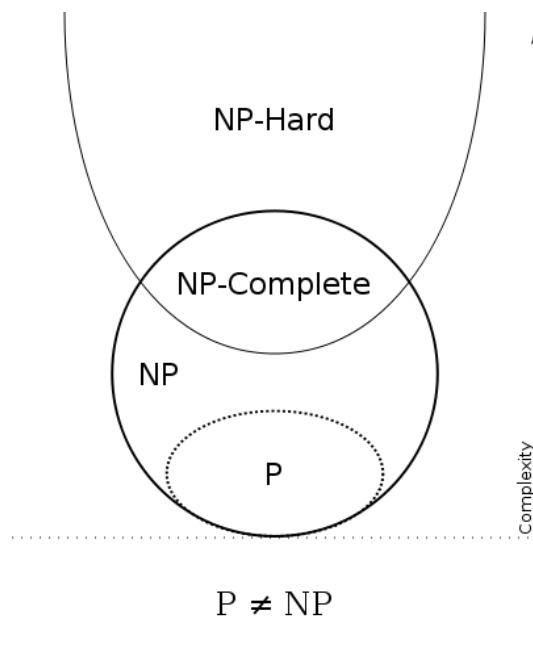
Της γενετικός αλγόριθμος (GA) είναι μια ευρετική αναζήτηση που είναι εμπνευσμένη από τη θεωρία της φυσικής εξέλιξης του Charles Darwin. Της ο αλγόριθμος αντικατοπτρίζει τη διαδικασία της φυσικής επιλογής όπου επιλέγονται τα πιο κατάλληλα άτομα για αναπαραγωγή προκειμένου να παραχθούν απόγονοι της επόμενης γενιάς. Η λύση της προβλήματος βελτιστοποίησης χρησιμοποιώντας τη μεθοδολογία GA περιλαμβάνει μια στοχαστική αναζήτηση του χώρου λύσης χρησιμοποιώντας σειρές ακεραίων, γνωστών ως χρωμοσώματα, που αντιπροσωπεύουν της παραμέτρους που βελτιστοποιούνται. Κάθε ακέραιος σε αυτά τα χρωμοσώματα είναι γνωστός ως γονίδιο και, για αυτές της εφαρμογές μοντελοποίησης, κάθε γονίδιο έχει μια δεκαδική τιμή μεταξύ 0 και 9. Θα πρέπει να σημειωθεί ότι αυτή δεν είναι η παραδοσιακή προσέγγιση GA όπου τα γονίδια είναι δυαδικές ποσότητες. Το πλεονέκτημα της δεκαδικής αναπαράστασης για αυτόν τον τύπο εφαρμογής είναι ότι επιτρέπει ένα ευρύτερο φάσμα πιθανών τιμών σε μικρότερα χρωμοσώματα και είναι ιδιαίτερα κατάλληλο για βελτιστοποίηση τόσο του μοντέλου όσο και του σχεδιασμού.[8], [11], [21]

2.8 Βελτιστοποίηση Σμήνους Σωματιδίων

Η βελτιστοποίηση σμήνους σωματιδίων (Particle Swarm Optimization = PSO) είναι ένας ισχυρός μετα-ευρετικός αλγόριθμος βελτιστοποίησης, εμπνευσμένος από τη συμπεριφορά σμήνους που παρατηρείται στη φύση, όπως το πέταγμα των πουλιών σε μορφή σμήνους προς αναζήτηση τροφής σε μια περιοχή. Το PSO είναι μια προσομοίωση ενός απλοποιημένου κοινωνικού συστήματος. Η αρχική πρόθεση του αλγορίθμου PSO ήταν να προσομοιώσει γραφικά τη χαριτωμένη αλλά απρόβλεπτη χορογραφία ενός σμήνους πουλιών. Με βάση τον αλγόριθμο, οι λύσεις αποτελούν τα σωματίδια του σμήνους τα οποία, ανταλλάσσοντας πληροφορίες μεταξύ τους, προοδευτικά μετακινούνται προς τη θέση που αντιστοιχεί στη βέλτιστη λύση.[12], [22], [23]

2.9 Υπολογιστική Πολυπλοκότητα

Στην επιστήμη των υπολογιστών, τα προβλήματα ανάλογα με την πολυπλοκότητά τους διαιρούνται σε κλάσεις. Η κλάση P (polynomial) αποτελείται από τα προβλήματα που επιλύονται σε πολυωνυμικό χρόνο, ενώ η κλάση NP περιλαμβάνει τα προβλήματα που μπορούν να επαληθευτούν εντός πολυωνυμικού χρόνου. Προφανώς, τα προβλήματα που ανήκουν στην κλάση P ανήκουν και στην NP ($P \subseteq NP$). Επίσης, έχουμε την κλάση NP-Complete (NP-Πλήρες) στην οποία ανήκουν τα προβλήματα που ανήκουν στην κλάση NP και όλα τα υπόλοιπα προβλήματα της κλάσης NP ανάγονται πολυωνυμικά σε αυτό. Το τελευταίο σύνολο προβλημάτων περιέχει τα δυσκολότερα και πιο πολύπλοκα προβλήματα στην επιστήμη των υπολογιστών, καθώς σε αυτή την κλάση βρίσκεται και το TDDT πρόβλημα που μελετάμε. Αυτά τα προβλήματα δεν είναι μόνο δύσκολο να λυθούν αλλά είναι και δύσκολο να επαληθευτούν, γι' αυτό το λόγο ονομάζονται NP-hard.



Εικόνα 14

Κάποια γνωστά NP-hard προβλήματα που συναντάμε στη βιβλιογραφία είναι:

- Το πρόβλημα τερματισμού (halting problem): είναι ένα πρόβλημα απόφασης, όπου δοθέντος ενός αυθαίρετου υπολογιστικού προγράμματος και μιας εισόδου, αποφασίζει αν αυτό τελικά θα σταματήσει εφόσον έχει τρέξει πεπερασμένες φορές ή αν θα συνεχίσει επ' άπειρον.
- Αληθείς ποσοτικοί Boolean τύποι (True Quantified Boolean Formulas): σημαίνει ότι έχουμε μια Boolean έκφραση και θέλουμε να μάθουμε εάν ορισμένες από τις μεταβλητές της μπορούν να εκφραστούν ως Boolean συνάρτηση για να μετατρέψουμε ολόκληρη την έκφραση σε ταυτολογία. Αυτό είναι σαν να λέμε: αν μια Boolean έκφραση με τις μεταβλητές και τις παραμέτρους της είναι Θεώρημα.[24]

Κεφάλαιο 3

3.1 Πειράματα

Μεταευρετική μέθοδος

Η εμπειρική λύση του προβλήματος χρονοπρογραμματισμού της υπεράσπισης της διατριβής υποδηλώνει ότι τα πιο περιορισμένα μέρη του προβλήματος είναι η ανάθεση των μαθητών σε συνεδρίες και η επιλογή των αντιπάλων τους. Αντίθετα, η εκπλήρωση των περιορισμών σύνθεσης της επιτροπής είναι συνήθως αρκετά εύκολη, χρησιμοποιώντας πρόσθετα διαθέσιμα μέλη. Για το λόγο αυτό, αποφασίζουμε να χρησιμοποιήσουμε μειωμένο χώρο αναζήτησης για τον τοπικό μας λύτη αναζήτησης, στον οποίο μόνο φοιτητές (και, κατά συνέπεια, επόπτες) και οι αντίπαλοι ανατίθενται σε συνεδρίες. Αυτό θα οδηγήσει ενδεχομένως σε μια επιτροπή που δεν ικανοποιεί τους περιορισμούς επικάλυψης (H3) και σύνθεσης H(5). Ωστόσο, αυτή η κατάσταση θα μπορούσε να ανακτηθεί στο τέλος της τοπικής αναζήτησης εκτελώντας ένα εποικοδομητικό βήμα μετα-επεξεργασίας που συμπληρώνει την επιτροπή με διαθέσιμα μέλη και ταυτόχρονα φροντίζει ώστε ο αριθμός των πιθανών καθηκόντων να ελαχιστοποιείται.

Αρχική λύση: Ο αλγόριθμος ξεκινά από μια λύση στην οποία αποτελείται από την ανάθεση όλων των φοιτητών σε οποιαδήποτε συνεδρία. Ο επιβλέπωντας και η επιλογή ενός από τους αντιπάλους εισάγονται στην επιτροπή του υποψηφίου. Αναλυτικότερα, βρίσκουμε αρχικά για ποιες συνεδρίες είναι διαθέσιμος κάθε καθηγητής και μαζί με τους αντίστοιχους φοιτητές, τους αναθέτουμε στις επιτρεπτές συνεδρίες τυχαία, αλλά επιλέγουμε τους καθηγητές (με τον αντίστοιχο φοιτητή) με αύξουσα σειρά ως προς το πλήθος των διαθέσιμων συνεδριών τους και κριτήριο το μέγιστο επιτρεπτό όριο φοιτητών για κάθε συνεδρία. Αυτό συμβαίνει, διότι π.χ. μπορεί ο τελευταίος φοιτητής που θα πρέπει να ανατεθεί σε κάποια από τις απομείναντες συνεδρίες (ή σε μία τελευταία συνεδρία) να μην είναι διαθέσιμος ο επιβλέπωντάς του. Επίσης, προσθέτουμε στις συνεδρίες έναν αντίπαλο από τους προτεινόμενους για τον κάθε φοιτητή, ή στην περίπτωση που ο φοιτητής δεν έχει προτεινόμενους αντιπάλους, όλοι είναι δηλαδή αντίπαλοι, δεν χρειάζεται να προσθέσουμε κάποιον επιπλέον. Αναφορικά με τους περιορισμούς, ο H1 είναι ικανοποιημένος από την κατασκευή, καθώς ο επιβλέπων «ακολουθεί» το φοιτητή στην επιτροπή, ο H4 είναι ικανοποιημένος με την αποτροπή αναθέσεων καθηγητών σε μη διαθέσιμες συνεδρίες και τέλος ο H2 ικανοποιείται αφού τηρείται το μέγιστο όριο των φοιτητών για κάθε συνεδρία. Δεδομένο ότι, μόνο φοιτητές (και, κατά συνέπεια, επόπτες) και οι αντίπαλοι ανατίθενται σε συνεδρίες, θα οδηγήσει ενδεχομένως σε μια επιτροπή που δεν ικανοποιεί τον περιορισμό σύνθεσης (H5), καθώς και για τη περίπτωση που έχουμε επικαλυπτόμενες συνεδρίες δεν ικανοποιείται ούτε ο περιορισμός H3. (Στην ανάθεση φοιτητών σε τυχαίες συνεδρίες, αναθέσαμε πρώτα τους φοιτητές που είναι πανομοιότυποι = με ίδιο επιβλέποντα και αντιπάλους, στην ίδια συνεδρία, και μετά συνεχίσαμε με τους υπόλοιπους)

Τελεστές γειτνίασης:

1. Αφαιρέστε έναν καθηγητή από μία συνεδρία.
2. Αναθέστε σε διαφορετική συνεδρία μια ομάδα φοιτητών (μαζί με τον επόπτη) που έχουν ανατεθεί στην ίδια συνεδρία και έχουν τον ίδιο επόπτη./
3. Αναθέστε έναν καθηγητή σε μία τυχαία συνεδρία (στην οποία είναι διαθέσιμος) .
4. Αναθέστε έναν φοιτητή σε διαφορετική συνεδρία και προσθέστε τον επιβλέποντά του και στην καινούρια συνεδρία (παραμένει στην παλιά συνεδρία).
5. Ανταλλάξτε συνεδρίες δύο ομάδων φοιτητών (μαζί και τους αντίστοιχους επόπτες), όπου οι φοιτητές κάθε ομάδας βρίσκονται στην ίδια συνεδρία και έχουν τον ίδιο επόπτη.

Αρχικά ο αλγόριθμος βρίσκει μία αρχική λύση και έπειτα εφαρμόζει επαναληπτικά τους πέντε τελεστές γειτνίασης. Οι πέντε αυτοί τελεστές έχουν την ίδια πιθανότητα να επιλεγούν, εφόσον ο αλγόριθμος διαλέγει τυχαία σε κάθε βήμα την κίνηση που θα κάνει.

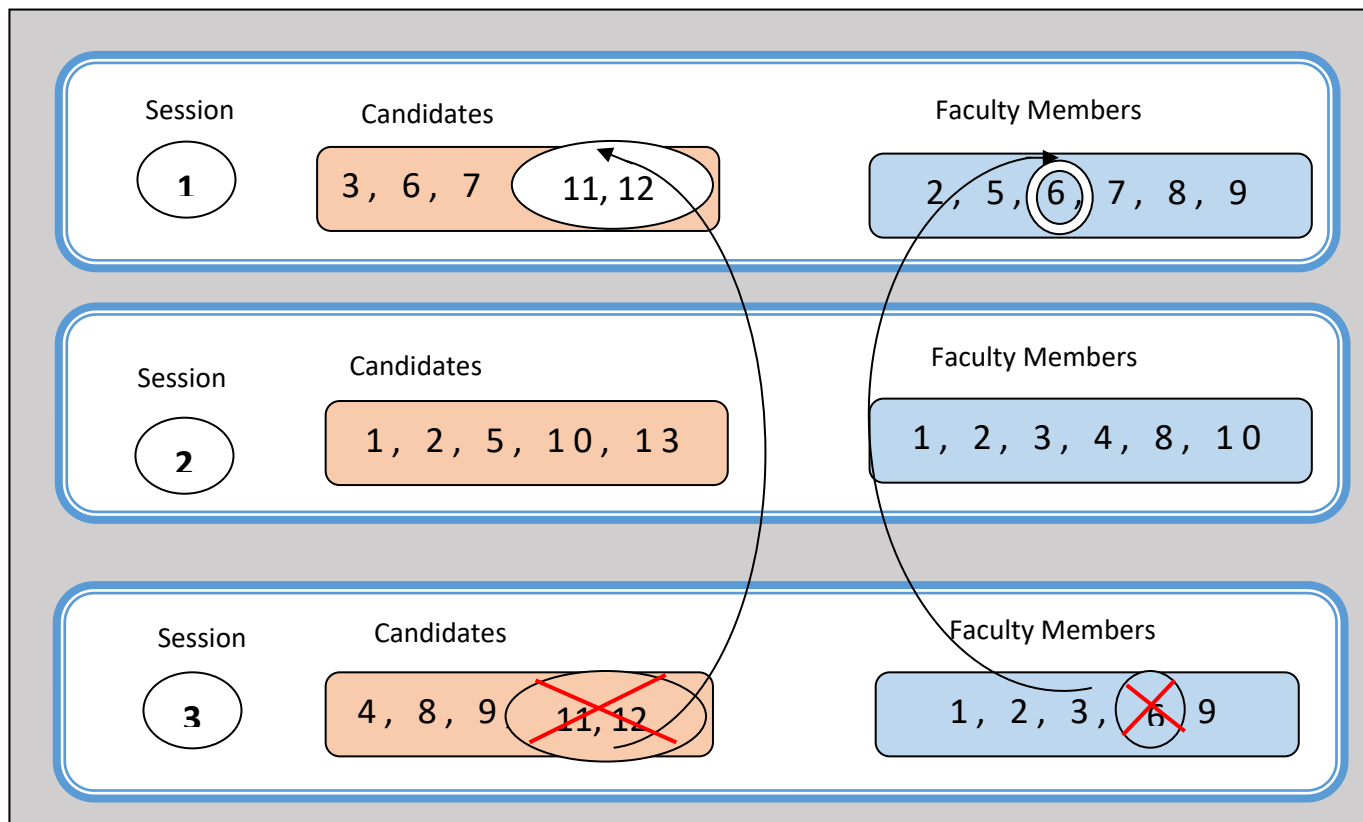
1. Αφαιρέστε έναν καθηγητή από μία συνεδρία.

Session 1	Candidates 3, 6, 7	Faculty Members 2, 5, 6, 7, 8, 9
Session 2	Candidates 1, 2, 5, 10, 13	Faculty Members 1, 2, 3 , 4, 8, 10
Session 3	Candidates 4, 8, 9, 11, 12	Faculty Members 1, 2, 3, 6, 9

Πίνακας 6: Αφαιρέστε έναν καθηγητή από μία συνεδρία

Ο αλγόριθμος διαλέγει τυχαία μία συνεδρία, επιλέγει πάλι τυχαία ένα μέλος της επιτροπής που βρίσκεται σε αυτή την συνεδρία και το διαγράφει. Στο παραπάνω παράδειγμα, επέλεξε την συνεδρία 2 και διέγραψε το μέλος 3.

2. Αναθέστε σε διαφορετική συνεδρία μια ομάδα φοιτητών (μαζί με τον επόπτη) που έχουν ανατεθεί στην ίδια συνεδρία και έχουν τον ίδιο επόπτη.



Πίνακας 7: Αναθέστε σε διαφορετική συνεδρία μια ομάδα φοιτητών (μαζί με τον επόπτη) που έχουν ανατεθεί στην ίδια συνεδρία και έχουν τον ίδιο επόπτη

Ο αλγόριθμος διαλέγει μία τυχαία συνεδρία (3) και βρίσκει τους αντίστοιχους φοιτητές που την απαρτίζουν (4, 8, 9, 11, 12). Μετά επιλέγει τυχαία ένα μέλος (6) αυτής της επιτροπής και βρίσκει τους αντίστοιχους φοιτητές (11, 12) που επιβλέπει (της συγκεκριμένης συνεδρίας). Διαλέγει άλλη μία τυχαία συνεδρία (1), με την προϋπόθεση να μην είναι πάλι αυτή που είχε επιλέξει πριν και να μην περιέχεται στο σύνολο των συνεδριών που δεν είναι διαθέσιμο το επιλεγμένο μέλος. Τέλος, μεταφέρει το επιλεγμένο μέλος (6) μαζί με τους επιβλεπόμενους φοιτητές του (11, 12) στην καινούρια συνεδρία (1) και τους αφαιρεί από την παλιά (3).

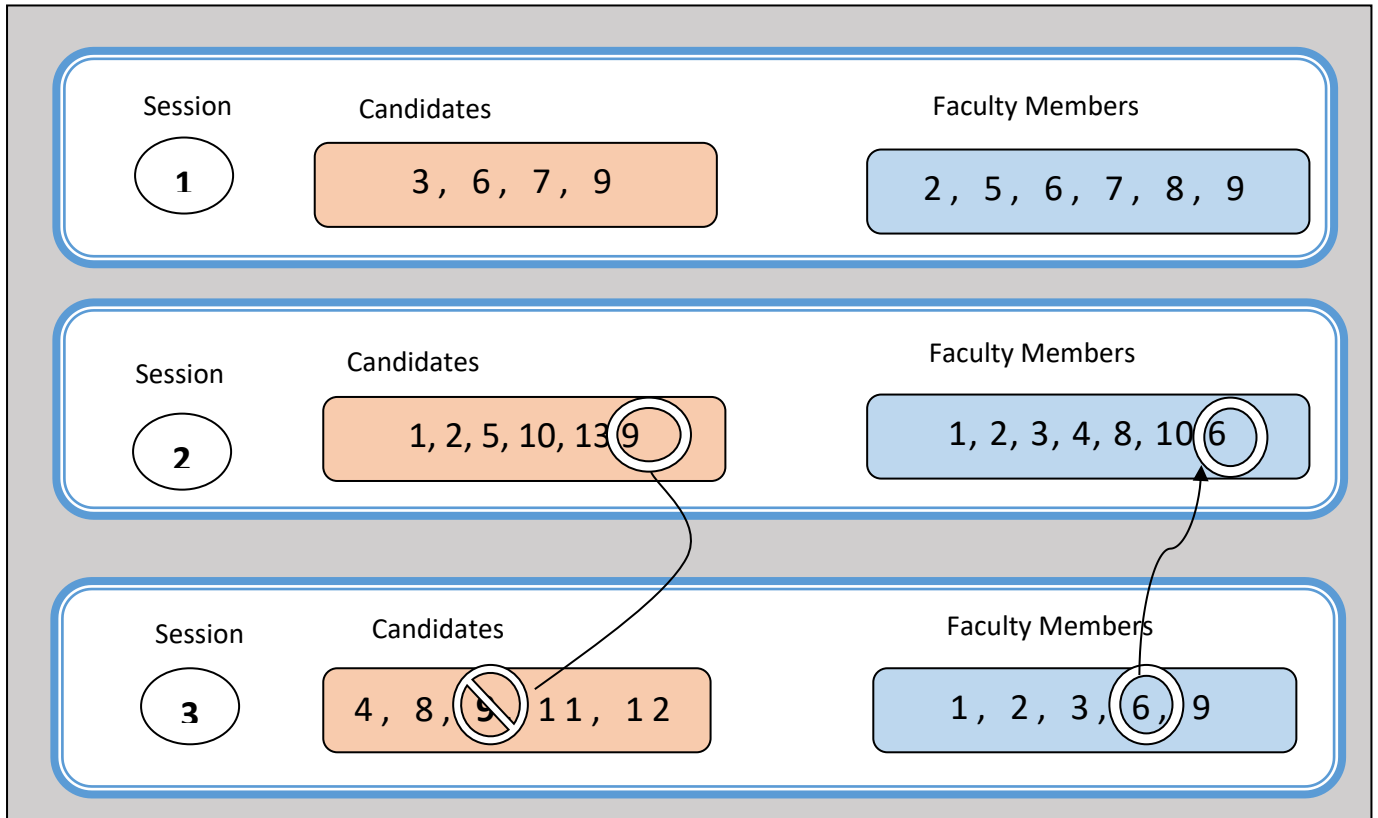
3. Αναθέστε έναν καθηγητή σε μία τυχαία συνεδρία (στην οποία είναι διαθέσιμος)

Session 1	Candidates 3, 6, 7	Faculty Members 9 2, 5, 6, 7, 8,
Session 2	Candidates 1, 2, 5, 10,	Faculty Members 1, 2, 3, 4, 8,
Session 3	Candidates 4, 8, 9, 11,	Faculty Members 1, 2, 3, 6, 9

Πίνακας 8: Αναθέστε έναν καθηγητή σε μία τυχαία συνεδρία (στην οποία είναι διαθέσιμος)

Σε αυτήν την περίπτωση, ο αλγόριθμος επιλέγει τυχαία μία συνεδρία (1) και βρίσκει τα μέλη αυτής της επιτροπής. Στη συνέχεια, επιλέγεται τυχαία ένα μέλος (9) από το σύνολο των μελών του προβλήματος (1, 2, 3,...,12), το οποίο δεν βρίσκεται ήδη στην επιτροπή αυτής της συνεδρίας καθώς και να είναι διαθέσιμο για παρακολούθηση της συνεδρίας αυτής.

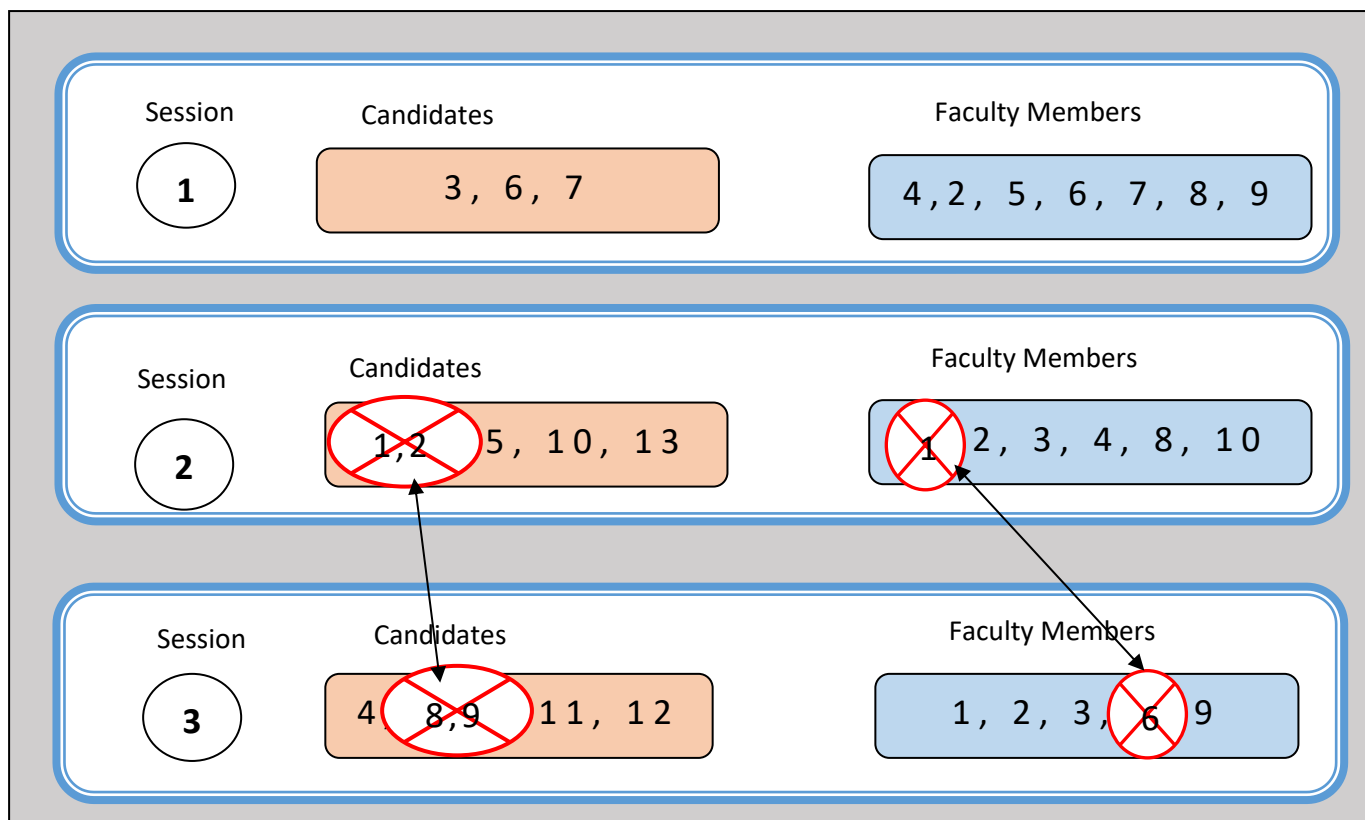
4. Αναθέστε έναν φοιτητή σε διαφορετική συνεδρία και προσθέστε τον επιβλέποντά του και στην καινούρια συνεδρία (παραμένει στην παλιά συνεδρία).



Πίνακας 9: Αναθέστε έναν φοιτητή σε διαφορετική συνεδρία και προσθέστε τον επιβλέποντά του και στην καινούρια συνεδρία (παραμένει στην παλιά συνεδρία)

Σε αυτόν τον τρόπο κίνησης, επιλέγεται τυχαία ένας υποψήφιος φοιτητής (π.χ. 9) από μία τυχαία συνεδρία (3) και εντοπίζεται ο αντίστοιχος επιβλέποντάς του (6). Έπειτα, επιλέγεται τυχαία μία δεύτερη συνεδρία (2), όχι ίδια με την πρώτη και στην οποία να είναι διαθέσιμος ο επιβλέπωντας, και τοποθετούνται αμφότεροι, φοιτητής και επιβλέπωντας, στη δεύτερη συνεδρία, ενώ διαγράφεται μόνο ο υποψήφιος φοιτητής από την παλιά.

5. Ανταλλάξτε συνεδρίες δύο ομάδων φοιτητών (μαζί και τους αντίστοιχους επόπτες), όπου οι φοιτητές κάθε ομάδας βρίσκονται στην ίδια συνεδρία και έχουν τον ίδιο επόπτη.



Πίνακας 10: Ανταλλάξτε συνεδρίες δύο ομάδων φοιτητών (μαζί και τους αντίστοιχους επόπτες), όπου οι φοιτητές κάθε ομάδας βρίσκονται στην ίδια συνεδρία και έχουν τον ίδιο επόπτη

Ο αλγόριθμος αρχικά βρίσκει τυχαία μία συνεδρία (3) με την αντίστοιχη επιτροπή μελών (1, 2, 3, 6, 9). Διαλέγει ένα μέλος αυτής της επιτροπής τυχαία (6), βρίσκοντας μαζί και τους αντίστοιχους φοιτητές που επιβλέπει (8, 9). Παρόμοια, επιλέγεται τυχαία μία δεύτερη συνεδρία (όχι η ίδια με πριν) (2), με ένα τυχαίο μέλος σε αυτή (1), και τους αντίστοιχους φοιτητές (1, 2) που το έχουν επιβλέποντα. Ανταλλάσσει το πρώτο σύνολο φοιτητών (8, 9) με το δεύτερο (1, 2) και τον πρώτο επιβλέποντα (6) με τον δεύτερο (1).

3.2 Μικτός Ακέραιος Προγραμματισμός

Χειρισμός του Τετραγωνικού Μέρους

Σε μια προσπάθεια περιορισμού των συνεδριών που ένα μέλος ΔΕΠ παρακολουθεί, ένα τετραγωνικό μέρος εισήχθη στο χαλαρό περιορισμό S1. Οι επιλυτές MIQP επιτρέπουν την ύπαρξη τετραγωνικών όρων στην αντικειμενική συνάρτηση, ενώ σε έναν επιλυτή MIP τέτοιοι όροι πρέπει πρώτα να γραμμικοποιηθούν.

Ο επιλυτής MIQP που έχει επιλεγεί, ο οποίος ήταν CPLEX, προέβλεψε μέσω μιας προηγμένης φάσης προ επεξεργασίας ότι θα ήταν καλύτερο να γραμμικοποιηθεί ο τετραγωνικός όρος και το έκανε αυτόματα. Η προσπάθειά μας να γραμμικοποιήσουμε τον τετραγωνικό όρο χρησιμοποιώντας τυπικές διαδικασίες οδήγησε στα ίδια ακριβώς αποτελέσματα με την αυτόματη μέθοδο που παρείχε ο CPLEX. Οι επιλυτές του προγραμματισμού με περιορισμούς συνήθως δεν μπορούν να χειριστούν τετραγωνικούς όρους, αλλά επιτρέπουν τον πολλαπλασιασμό μεταξύ όρων συμπεριλαμβανομένων των μεταβλητών απόφασης. Χρησιμοποιήσαμε το εργαλείο επίλυσης CPSat OR Tools της Google που υποστηρίζει τη μέθοδο που αντιγράφει την επιθυμητή συμπεριφορά. Ως εκ τούτου, το παραπάνω μοντέλο μπορεί να χρησιμοποιηθεί σε όλους τους επιλυτές MIP, MIQP και CP.

Το αρχικό μοντέλο, που περιγράφεται παραπάνω, μπορεί να είναι σε θέση να λύσει μέτριες περιπτώσεις προβλήματος του συνόλου δεδομένων. Για μεγάλες και δύσκολες περιπτώσεις προβλημάτων μπορούμε να τροποποιήσουμε αυτό το μοντέλο και να δημιουργήσουμε μια κατά προσέγγιση έκδοση. Η ιδέα πίσω από την προσέγγιση είναι να εκμεταλλευτεί το γεγονός ότι οι υποψήφιοι δεν συμμετέχουν σε κανένα αυστηρό περιορισμό εκτός από τον αυστηρό περιορισμό H1 που εξασφαλίζει ότι ένας υποψήφιος πρέπει να προγραμματιστεί σε μια συνεδρία όπου ο επιβλέπων του είναι παρών. Επιλέγουμε να αγνοήσουμε τον χαλαρό περιορισμό και να παράγουμε ένα βελτιστοποιημένο πρόγραμμα όπου υπάρχει χώρος για όλους τους υποψηφίους να είναι παρόντες μαζί με τους επιβλέποντές τους. Χρησιμοποιώντας αυτό το κόλπο, είμαστε σε θέση να αγνοήσουμε τον χαλαρό περιορισμό δεδομένου ότι όλοι οι υποψήφιοι του ίδιου προϊσταμένου είναι πρακτικά όμοιοι και συνεισφέρουν μόνο ως μονάδες που μπορούν να θεωρηθούν ανταλλάξιμες.

Κάποια σημαντικά μέρη του κώδικα που αξίζει να παρουσιάσουμε είναι:

➤ *Η υλοποίηση των αυστηρών περιορισμών*

Έχει δημιουργηθεί μία κλάση problem που περιλαμβάνει τα δεδομένα κάποιου στιγμιότυπου και φτιάχνουμε τους παρακάτω περιορισμούς, όπως τους έχουμε ορίσει στο κεφάλαιο 1:

- **H1. Επίβλεψη:** Για κάθε φοιτητή, ο επιβλέπων καθηγητής πρέπει να είναι παρών στη συνεδρία που έχει ανατεθεί στον φοιτητή.

$$x_{cs} \leq y_{Fs} \quad \forall c \in \mathbb{C}, \forall s \in \mathbb{S}$$

```

## H1. Supervision
for candidate in problem.all_candidates:
    for session in sessions_range:
        faculty_member = problem.supervisor[candidate]
        model.add(
            x_variables[(candidate, session)]
            <= y_variables[(faculty_member, session)]
        )

```

- **H2. Φοιτητές ανά συνεδρία:** Ο αριθμός των φοιτητών που αναθέτονται σε μια συνεδρία δεν πρέπει να υπερβαίνει ένα δεδομένο μέγιστο.

$$\sum_{c \in \mathbb{C}} x_{cs} \leq C \quad \forall s \in \mathbb{S}$$

```

## H2. Students Per Session
for session in sessions_range:
    model.add(
        sum(
            x_variables[(candidate, session)]
            for candidate in problem.all_candidates
        )
        <= problem.max_candidates
    )

```

- **H3. Επικάλυψη συνεδριών:** Σε περίπτωση που δύο συνεδρίες γίνονται την ίδια στιγμή, ένα μέλος της σχολής μπορεί να ανατεθεί το πολύ σε μία από τις δύο επιτροπές.

$$y_{fs_1} + y_{fs_2} \leq 1 \quad \forall s_1, s_2 \in \mathbb{Q}, \forall f \in \mathbb{F}$$

```

## H3. Overlapping Sessions
for session1, session2 in problem.simultaneous_sessions:
    for faculty_member in problem.all_faculty_members:
        model.add(
            y_variables[(faculty_member, session1)]
            + y_variables[(faculty_member, session2)]
            <= 1
        )

```

- **H4. Διαθεσιμότητα:** Κάθε μέλος της επιτροπής πρέπει να είναι διαθέσιμο στη συνεδρία που έχει προγραμματιστεί.

$$y_{fs} = 0 \quad \forall f \in \mathbb{F}, \forall s \in \mathbb{S}_f$$

```
## H4. Availability
for faculty_member, sessions in problem.unavailability.items():
    for session in sessions:
        model.add(y_variables[(faculty_member, session)] == 0)
        # Additional constraint
        # Force a candidate not to attend a session where his supervisor is
unavailable
        if faculty_member in problem.supervises:
            for candidate in problem.supervises[faculty_member]:
                model.add(x_variables[(candidate, session)] == 0)
```

- **H5. Σύνθεση επιτροπής:** Οι συνεδρίες πρέπει να σχηματίζονται τηρώντας έναν ελάχιστο και μέγιστο αριθμό προσωπικού ΔΕΠ που κατέχει συγκεκριμένο ακαδημαϊκό επίπεδο.

$$m_l \leq \sum_{f \in \mathbb{F}, A_f \leq 1} y_{fs} \leq M_l \quad \forall l \in \mathbb{L}, \forall s \in \mathbb{S}$$

```
## H5. Committee Composition

for level, bounds in problem.limit_members_for_level.items():
    for session in sessions_range:
        min_faculty_members, max_faculty_members = bounds
        model.add(
            sum(
                y_variables[(faculty_member, session)]
                for faculty_member, faculty_level in problem.academic_level.items()
                if faculty_level <= level
            )
            >= min_faculty_members
        )
        model.add(
            sum(
                y_variables[(faculty_member, session)]
                for faculty_member, faculty_level in problem.academic_level.items()
                if faculty_level <= level
            )
            <= max_faculty_members
        )
```

➤ Η υλοποίηση των χαλαρών περιορισμών:

S1. Πολλαπλά Καθήκοντα: Κάθε φορά που ένα μέλος ΔΕΠ συμμετέχει σε περισσότερες από μία επιτροπές, επιβάλλεται ποινή. Εάν a είναι ο αριθμός των παρουσιών ενός μέλους ΔΕΠ, η σχετική ποινή είναι $(a-1)^2$ για $a > 0$, και 0 για $a = 0$.

$$\sum_{f \in \mathbb{Q}_c} (y_{fs}) + z_c \geq x_{fs} \quad \forall c \in \mathbb{C}, |\mathbb{Q}_c| \neq 0, \forall s \in \mathbb{S}$$

```
## Force z_variables

for candidate in z_variables:
    for session in sessions_range:
        if session in
problem.unavailability[problem.supervisor[candidate]]:
            continue
        model.add(
            sum(
                y_variables[(faculty_member, session)]
                for faculty_member in problem.opponents[candidate]
            )
            + z_variables[candidate]
            >= x_variables[(candidate, session)]
        )
```

S2. Παρουσία αντιπάλου: Επιβάλλεται ποινή εάν ένας υποψήφιος παρευρεθεί σε μια συνεδρία χωρίς να είναι παρών κανένας από τους κατάλληλους αντιπάλους του. Εάν ο υποψήφιος δεν έχει κατάλληλους αντιπάλους δεν επιβάλλεται ποινή.

$$\sum_{s \in \mathbb{S}} (y_{fs}) - 1 \leq u_f \quad \forall f \in \mathbb{F}$$

```
## Force v_variables

for faculty_member in problem.all_faculty_members:
    model.add(
        sum([y_variables[(faculty_member, session)] for session in sessions_range])
        - 1
        <= v_variables[faculty_member]
    )
```

➤ Η υλοποίηση της αντικειμενικής συνάρτησης:

$$\min w_{s1} * \sum_{f \in F} u_f^2 + w_{s2} * \sum_{c \in C, |Q_c| \neq 0} z_c$$

```
# Objective Function

model.minimize(
    problem.ws1
    * sum(
        v_variables[faculty_member] ** 2
        for faculty_member in problem.all_faculty_members
    )
    + problem.ws2 * sum(z_variables[candidate] for candidate in z_variables)
)
```

3.4 Αποτελέσματα

Τα σημεία αναφοράς λειτουργούσαν σε Intel Xeon E5-2630@2,4Ghz, με 48GB RAM (AMD A4-5000 APU with Radeon(TM) HD Graphics 1.50 GHz, με 4,00 GB RAM). Ο κώδικας προγραμματίστηκε χρησιμοποιώντας Python API των επιλυτών που χρησιμοποιήθηκαν (CPLEX και CP Sat) και δόθηκε χρόνος εκτέλεσης 542 δευτερολέπτων για κάθε περίπτωση προβλήματος, δηλαδή το χρονικό όριο που χρησιμοποιείται στο [1]. Τα πρώτα 30 δευτερόλεπτα ήταν αφιερωμένα στο μοντέλο μηδενικής ποινής και τα επόμενα 30 δευτερόλεπτα στο κατά προσέγγιση μοντέλο. Ο υπόλοιπος χρόνος εκτέλεσης ήταν ο χρόνος εκτέλεσης για το πλήρες μοντέλο.

Έχουμε χρησιμοποιήσει δύο λύτες, το CP Sat από το Google ORTools και το CPLEX από την IBM ILOG και έχουμε προγραμματίσει και τα τρία μοντέλα που περιγράφονται παραπάνω. Ο Πίνακας 11 δείχνει τα αποτελέσματα της εφαρμογής της δικής μας προσέγγισης τριών φάσεων στις πραγματικές περιπτώσεις TDDT που προήλθαν από ιταλικά πανεπιστήμια. Η στήλη "Best known" εμφανίζει τις καλύτερες τιμές μεταξύ των 3 προσεγγίσεων (CP, IP και Τοπική αναζήτηση) που χρησιμοποιήθηκαν στο [1]. Για κάθε περίπτωση προβλήματος υπολογίζουμε τη συνολική καλύτερη τιμή που επιτυγχάνεται και αυτή σημειώνεται με έντονους χαρακτήρες. Ορίζουμε ως «Error» μια τιμή που είναι το ποσοστό ότι αυτή η τιμή είναι

χειρότερη από την καλύτερη. Δύο στήλες του πίνακα 11 παρουσιάζουν τιμές σφάλματος για τους δύο λύτες που χρησιμοποιήθηκαν. Ο λύτης CP Sat επιστρέφει, σχεδόν πάντα, καλύτερα αποτελέσματα από τις άλλες εναλλακτικές λύσεις. Το πρόγραμμα επίλυσης CPLEX καταφέρνει να επιτύχει λύσεις για όλες τις περιπτώσεις προβλημάτων, επιλύει προβλήματα που έχουν μηδενικό κόστος ποινής, αλλά έχει μάλλον κακή απόδοση για τις υπόλοιπες περιπτώσεις προβλημάτων.

Ο Πίνακας 12 παρουσιάζει αποτελέσματα για το τεχνητό σύνολο περιπτώσεων προβλημάτων. Σημειώστε ότι το παράδειγμα προβλήματος "art25" είναι γνωστό ότι δεν είναι εφικτό, επομένως δεν παρουσιάζονται αποτελέσματα σε αυτήν τη σειρά. Οι περισσότερες βέλτιστες τιμές επιτυγχάνονται από το CP Sat, ενώ το CPLEX είναι σε θέση να επιτύχει τιμές για όλες τις περιπτώσεις προβλημάτων, κάτι που δεν συνέβη για τον επιλυτή IP στο [1] που άφησε 8 παρουσίες προβλημάτων χωρίς να παράγει μια εφικτή λύση.

Instance	Best known	CP Sat	Error (CP Sat)	CPLEX	Error (CPLEX)
real01	36	36	0.00	41	0.14
real02	0	0	0.00	0	0.00
real03	49	49	0.00	57	0.16
real04	25	25	0.00	84	2.36
real05	0	0	0.00	0	0.00
real06	0	0	0.00	0	0.00
real07	31	32	0.03	41	0.32
real08	17	17	0.00	17	0.00
real09	42	35	0.00	167	3.77
real10	40	41	0.03	173	3.33
real11	13	11	0.00	71	5.45
real12	34	35	0.03	257	6.56
real13	0	0	0.00	0	0.00
real14	1	1	0.00	10	9.00
real15	306	291	0.00	1075	2.69
real16	7	7	0.00	19	1.71
real17	21	21	0.00	21	0.00
real18	46	46	0.00	69	0.50
real19	0	0	0.00	0	0.00
real20	17	17	0.00	49	1.88
real21	9	9	0.00	9	0.00

Πίνακας 11: Αποτελέσματα της εφαρμογής τριών φάσεων στις πραγματικές περιπτώσεις για CP Sat και CPLEX

Instance	Best known	CP Sat	Error (CP Sat)	CPLEX	Error (CPLEX)
art01	82	87	0.06	581	6.09
art02	16	16	0.00	16	0.00
art03	1	1	0.00	46	45.00
art04	0	0	0.00	0	0.00
art05	0	0	0.00	0	0.00
art06	110	117	0.06	885	7.05
art07	201	202	0.00	1122	4.58
art08	0	0	0.00	0	0.00
art09	8	8	0.00	8	0.00
art10	12	12	0.00	12	0.00
art11	215	212	0.00	986	3.65
art12	0	0	0.00	0	0.00
art13	78	78	0.00	78	0.00
art14	36	29	0.00	986	33.00
art15	169	181	0.07	1149	5.80
art16	63	62	0.00	1141	17.40
art17	0	0	0.00	0	0.00
art18	0	0	0.00	0	0.00
art19	0	0	0.00	0	0.00
art20	12	0	0.00	0	0.00
art21	0	0	0.00	0	0.00
art22	0	0	0.00	0	0.00
art23	171	172	0.01	539	2.15
art24	0	0	0.00	0	0.00
art25	-	-	-	-	-
art26	14	14	0.00	34	1.43
art27	219	229	0.05	1272	4.81
art28	0	0	0.00	0	0.00
art29	203	196	0.00	951	3.85
art30	0	0	0.00	0	0.00
art31	26	20	0.00	569	27.45
art32	0	0	0.00	0	0.00
art33	0	0	0.00	0	0.00
art34	0	0	0.00	0	0.00
art35	0	0	0.00	0	0.00
art36	0	0	0.00	0	0.00
art37	0	0	0.00	0	0.00
art38	1	1	0.00	51	50.00
art39	11	10	0.00	11	0.10
art40	9	12	0.33	12	0.33
art41	0	0	0.00	0	0.00
art42	4	4	0.00	4	0.00
art43	0	0	0.00	0	0.00
art44	0	0	0.00	0	0.00
art45	7	7	0.00	7	0.00

Πίνακας 12: Αποτελέσματα της εφαρμογής τριών φάσεων στις τεχνητές περιπτώσεις για CP Sat και CPLEX

Παρακάτω έχουμε επίσης τα αποτελέσματα που βγάζει η εφαρμογή της Προσομοιωμένης Ανόπτησης για τα πραγματικά στιγμιότυπα στον Πίνακα 13 και για τα τεχνητά στιγμιότυπα στον Πίνακα 14 σε σύγκριση με τις καλύτερες λύσεις που έχουν βρεθεί (στον Πίνακα 12).

Στη στήλη Infeasibilities έχουμε τον αριθμό παραβιάσεων των αυστηρών περιορισμών, γι' αυτόν το λόγο όταν υπάρχει παραβίαση δεν υπάρχει νόημα για το αποτέλεσμα που βγάζει η εφαρμογή και έτσι δεν αναγράφεται. Γίνεται αντιληπτό ότι η συγκεκριμένη εφαρμογή σε σύγκριση με τους προηγούμενες επιλυτές CPLEX και CP Sat έχει τις χειρότερες λύσεις για όλα τα στιγμιότυπα που μελετήθηκαν.

Instance	Infeasibilities	Simulated Annealing	Best solution
real01	0	283	36
real02	0	77	0
real03	0	428	49
real04	0	489	25
real05	1	-	0
real06	0	183	0
real07	0	194	31
real08	0	47	17
real09	0	524	35
real10	0	770	40
real11	0	514	11
real12	0	691	34
real13	0	162	0
real14	0	288	1
real15	0	1905	291
real16	0	333	7
real17	0	30	21
real18	0	479	46
real19	0	153	0
real20	0	400	17
real21	0	31	9

Πίνακας 13: Αποτελέσματα εφαρμογής με Προσομοιωμένη Ανόπτηση για τις πραγματικές περιπτώσεις

Instance	Infeasibilities	Simulated Annealing	Best solution
art01	0	1214	82
art02	0	95	16
art03	0	632	1
art04	0	1194	0
art05	0	396	0
art06	0	1532	110
art07	0	1843	201
art08	0	1486	0
art09	0	169	8
art10	0	104	12
art11	0	1784	212
art12	0	804	0
art13	0	364	78
art14	1	-	29
art15	1	-	169
art16	0	1822	62
art17	2	-	0
art18	0	779	0
art19	0	519	0
art20	11	-	0
art21	0	1456	0
art22	0	944	0
art23	0	1110	171
art24	0	70	0
art25	2	-	-
art26	0	581	14
art27	3	-	219
art28	0	1010	0
art29	0	1610	196
art30	0	1155	0
art31	0	1075	20
art32	0	215	0
art33	0	493	0
art34	0	138	0
art35	0	347	0
art36	0	364	0
art37	0	266	0
art38	0	642	1
art39	0	1149	10
art40	0	973	9
art41	0	342	0
art42	0	586	4
art43	0	496	0

art44	0	286	0
art45	0	869	7

Πίνακας 14: Αποτελέσματα εφαρμογής με Προσομοιωμένη Ανόπτηση για τις τεχνητές περιπτώσεις

Κεφάλαιο 4

Συμπέρασμα

Το πρόβλημα του χρονοδιαγράμματος υπεράσπισης της διατριβής είναι ένα πρόβλημα που πολλά πανεπιστήμια πρέπει να αντιμετωπίσουν σε κάθε ακαδημαϊκό έτος. Όταν εμπλέκονται μεγάλοι αριθμοί φοιτητών, μελών επιτροπής και συνεδριών, το πρόβλημα γίνεται σύντομα δύσκολο. Γίνεται ακόμη πιο δύσκολο με την παρουσία αρκετών κανονισμών και περιορισμών που επιβάλλονται από τη διοίκηση. Η μελέτη μας αφορούσε μια συγκεκριμένη παραλλαγή αυτού του προβλήματος που ευτυχώς έχει έναν σαφή ορισμό και ένα δημοσίως διαθέσιμο σύνολο δεδομένων. Στην προσπάθειά μας να προσεγγίσουμε το πρόβλημα διατυπώσαμε μαθηματικά το πρόβλημα και χρησιμοποιήσαμε μαθηματικούς λύτες προγραμματισμού, προγραμματισμό υπό περιορισμούς και Τοπική Αναζήτηση. Λόγω των μεγάλων μεγεθών των μοντέλων που προέκυψαν, οι λύτες δεν μπόρεσαν να λύσουν άμεσα ορισμένα από τα προβλήματα. Έτσι, χρησιμοποιήσαμε μια προσέγγιση τριών φάσεων που αρχικά προσπαθεί να αποδείξει ότι υπάρχει μια βέλτιστη λύση μηδενικού κόστους.

Εάν συμβαίνει αυτό, τότε βρίσκεται η βέλτιστη λύση και η διαδικασία σταματά. Διαφορετικά, χρησιμοποιείται μια δεύτερη κατά προσέγγιση διατύπωση που προσπαθεί να επιτύχει καλύτερες τιμές για το τετραγωνικό μέρος του στόχου. Αυτή η φάση είναι ικανή να βρει μια εφικτή λύση γρήγορα για όλες τις περιπτώσεις προβλημάτων του συνόλου δεδομένων χρησιμοποιώντας είτε τον επιλυτή ILOG CPLEX της IBM είτε τον επιλυτή CPSat ORTools. Η τελική φάση τροφοδοτεί το πλήρες μοντέλο του προβλήματος με μια εφικτή λύση που έχει βρει η προηγούμενη φάση και στη συνέχεια ειδικά ο λύτης ORTools CPSat είναι σε θέση να επιτύχει λύσεις υψηλής ποιότητας σε μέτριο χρόνο. Τα αποτελέσματα που επιτυγχάνονται από την προσέγγισή μας είναι συγκρίσιμα και, σε πολλές περιπτώσεις, καλύτερα από τα πιο γνωστά αποτελέσματα, όπως φαίνεται στις ενότητες αποτελεσμάτων της εργασίας.

Τέλος, έγινε μία προσέγγιση του προβλήματος με Προσομοιωμένη Ανόπτηση, όπου τα αποτελέσματα ήταν υποδεέστερα, αλλά θα μπορούσε να εφαρμοστεί μελλοντικά ίσως σε κάποια παραλλαγή του προβλήματος ή σε μεγαλύτερα προβλήματα.

Βιβλιογραφία

- [1] M. Battistutta, S. Ceschia, F. De Cescio, L. Di Gaspero, και A. Schaerf, ‘Modelling and solving the thesis defense timetabling problem’, *Journal of the Operational Research Society*, τ. 70, τχ. 7, σσ. 1039–1050, Ιουλίου 2019, doi: 10.1080/01605682.2018.1495870.
- [2] S. Ceschia, L. Di Gaspero, και A. Schaerf, ‘Educational timetabling: Problems, benchmarks, and state-of-the-art results’, *European Journal of Operational Research*, Ιουλίου 2022, doi: 10.1016/j.ejor.2022.07.011.
- [3] J. H. Kingston, ‘Educational Timetabling’, στο *Automated Scheduling and Planning: From Theory to Practice*, A. S. Uyar, E. Ozcan, και N. Urquhart, Επιμ. Berlin, Heidelberg: Springer, 2013, σσ. 91–108. doi: 10.1007/978-3-642-39304-4_4.
- [4] A. Schaerf, ‘A Survey of Automated Timetabling’, σ. 42.
- [5] R. Qu, E. Burke, B. Mccollum, L. Merlot, και S. Lee, ‘A Survey of Search Methodologies and Automated Approaches for Examination Timetabling’, *Journal of Scheduling - SCHEDULING*, Ιανουαρίου 2006.
- [6] J. S. Tan, S. L. Goh, G. Kendall, και N. R. Sabar, ‘A survey of the state-of-the-art of optimisation methodologies in school timetabling problems’, *Expert Systems with Applications*, τ. 165, σ. 113943, Μαρτίου 2021, doi: 10.1016/j.eswa.2020.113943.
- [7] L. T. Bui και V. Hoang, ‘A Multi-Objective Approach for Master’s Thesis Committees Scheduling Using DMEA’, στο *Simulated Evolution and Learning*, τ. 7673, L. T. Bui, Y. S. Ong, N. X. Hoai, H. Ishibuchi, και P. N. Suganthan, Επιμ. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, σσ. 450–459. doi: 10.1007/978-3-642-34859-4_45.
- [8] T. T. B. Huynh, Q. D. Pham, και D. D. Pham, ‘Genetic Algorithm for Solving the Master Thesis Timetabling Problem with Multiple Objectives’, στο *2012 Conference on Technologies and Applications of Artificial Intelligence*, Tainan, Taiwan, Νοεμβρίου 2012, σσ. 74–79. doi: 10.1109/TAAL.2012.50.
- [9] B. Kochaniková και H. Rudová, ‘Student Scheduling for Bachelor State Examinations’, σ. 5.
- [10] P. Dung, H. Trung, T. Hoang, και N. Hoang, ‘A Java library for Constraint-Based Local Search: Application to the master thesis defense timetabling problem’, Δεκεμβρίου 2015, σσ. 1–8. doi: 10.1145/2833258.2833300.
- [11] S. Limanto, N. Benarkah, και T. Adelia, ‘Thesis Examination Timetabling using Genetic Algorithm’, στο *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, Bali, Indonesia, Οκτωβρίου 2018, σσ. 6–10. doi: 10.1109/KCIC.2018.8628572.
- [12] G. Christopher και A. Wicaksana, ‘Particle swarm optimization for solving thesis defense timetabling problem’, *TELKOMNIKA*, τ. 19, τχ. 3, σ. 762, Ιουνίου 2021, doi: 10.12928/telkomnika.v19i3.18792.
- [13] J. Almeida, D. R. dos Santos, και J. R. Figueira, ‘A Multi-Objective Model for Thesis Defence Scheduling’. arXiv, 16 Μάιος 2022. Ημερομηνία πρόσβασης: 13 Αύγουστος 2022. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <http://arxiv.org/abs/2205.07727>
- [14] G. M. Thompson, ‘Improving conferences through session scheduling’, *The Cornell Hotel and Restaurant Administration Quarterly*, τ. 43, τχ. 3, σσ. 71–76, Ιουνίου 2002, doi: 10.1016/S0010-8804(02)80020-9.
- [15] R. Correia, A. Subramanian, T. Bulhões, και P. H. V. Penna, ‘Scheduling the Brazilian OR conference’, *Journal of the Operational Research Society*, τ. 73, τχ. 7, σσ. 1487–1498, 2022.
- [16] L. A. Wolsey, ‘Mixed Integer Programming’, στο *Wiley Encyclopedia of Computer Science and Engineering*, John Wiley & Sons, Ltd, 2008, σσ. 1–10. doi: 10.1002/9780470050118.ecse244.
- [17] V. Kumar και S. M. Yadav, ‘A state-of-the-Art review of heuristic and metaheuristic optimization techniques for the management of water resources’, *Water Supply*, τ. 22, τχ. 4, σσ. 3702–3728, Ιανουαρίου 2022, doi: 10.2166/ws.2022.010.

- [18]P. Alefragis, C. Gogos, C. Valouxis, και E. Housos, ‘A multiple metaheuristic variable neighborhood search framework for the Uncapacitated Examination Timetabling Problem’, σ. 13.
- [19]E. Aarts, J. Korst, και W. Michiels, ‘Simulated Annealing’, στο *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*, E. K. Burke και G. Kendall, Επιμ. Boston, MA: Springer US, 2005, σσ. 187–210. doi: 10.1007/0-387-28356-0_7.
- [20]R. A. Rutenbar, ‘Simulated annealing algorithms: an overview’, *IEEE Circuits and Devices Magazine*, τ. 5, τχ. 1, σσ. 19–26, Ιανουαρίου 1989, doi: 10.1109/101.17235.
- [21]F.-S. Wang και L.-H. Chen, ‘Genetic Algorithms’, στο *Encyclopedia of Systems Biology*, W. Dubitzky, O. Wolkenhauer, K.-H. Cho, και H. Yokota, Επιμ. New York, NY: Springer New York, 2013, σσ. 819–820. doi: 10.1007/978-1-4419-9863-7_412.
- [22]J. Kennedy και R. Eberhart, ‘Particle swarm optimization’, στο *Proceedings of ICNN’95 - International Conference on Neural Networks*, Αυγούστου 1995, τ. 4, σσ. 1942–1948 τ.4. doi: 10.1109/ICNN.1995.488968.
- [23]G. Koulinas, L. Kotsikas, και K. Anagnostopoulos, ‘A particle swarm optimization based hyper-heuristic algorithm for the classic resource constrained project scheduling problem’, *Information Sciences*, τ. 277, σσ. 680–693, Σεπτεμβρίου 2014, doi: 10.1016/j.ins.2014.02.155.
- [24]S. Naseera, ‘P, NP, NP-Hard & NP-complete problems’, σ. 46.